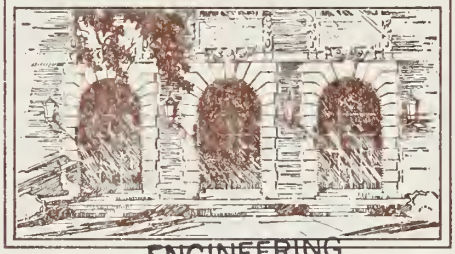
LIBRARY OF THE
UNIVERSITY OF ILLINOIS
AT URBANA-CHAMPAIGN

510.84
I163c
no.51-60



ENGINEERING

AUG 5 1976

The person charging this material is responsible for its return to the library from which it was withdrawn on or before the **Latest Date** stamped below.

Theft, mutilation, and underlining of books are reasons for disciplinary action and may result in dismissal from the University.

UNIVERSITY OF ILLINOIS LIBRARY AT URBANA-CHAMPAIGN

ENGINEERING

CONFERENCE ROOM

OCT 18 1988

OCT 11 RECD

MAY 3 1990

MAY 7 RECD

FEB 27 1991

MAR 05 RECD

L161—O-1096

510.84
I163c
no.58

Engin.

CONFERENCE ROOM

ENGINEERING LIBRARY
UNIVERSITY OF ILLINOIS
URBANA, ILLINOIS

Center for Advanced Computation

UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN
URBANA, ILLINOIS 61801

CAC Document No. 58

FAST HEURISTIC TECHNIQUES FOR PLACING AND WIRING PRINTED CIRCUIT BOARDS

By

James Edward Stevens, Jr.

October 1972

CAC Document No. 58

FAST HEURISTIC TECHNIQUES FOR PLACING AND WIRING PRINTED CIRCUIT BOARDS

By

James Edward Stevens, Jr.

Center for Advanced Computation
University of Illinois at Urbana-Champaign
Urbana, Illinois 61801

October 1972

Submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Computer Science in the Graduate College of the University of Illinois at Urbana-Champaign and supported in part by the Advanced Research Projects Agency of the Department of Defense and was monitored by the U.S. Army Research Office-Durham under Contract No. DAHC04 72-C-0001.


Digitized by the Internet Archive
in 2012 with funding from
University of Illinois Urbana-Champaign

http://archive.org/details/fastheuristictec58stev

ENGINEERING LIBRARY

ABSTRACT

A comparison of existing placement techniques is presented. Large realistic printed circuit board problems are introduced and used to make comparisons. The value of different placements is measured by the total wire length generated as well as by the results of actually routing the connections on the board.

A new wire routing algorithm called Channel Routing, is proposed. The Channel Routing algorithm is faster than existing routing techniques and can handle very large circuit board problems. The results of implementing the basic Channel Routing algorithms are presented. These results are also used to compare the effectiveness of the different placement algorithms.

ACKNOWLEDGEMENT

I am indebted to Professor D. L. Slotnick for his support and encouragement and to Dr. Akihiro Hashimoto for his valuable advice.

The typing was done by Frieda Anderson and the drafting by Fred Hancock.

Credit is also due Linda Zapf for programming the placement algorithms.

I am grateful to the Center for Advanced Computation and the Department of Computer Science at the University of Illinois for their support of this dissertation.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1. INTRODUCTION

Since its beginning in 1955, automated design of digital electronic equipment has become a formidable area of study. A large body of information has been published in the open literature [3], including a recently published book, [11], which deals solely with the current techniques for the automated design of computer hardware. Design automation has been incorporated into every level of manufacturing, from register level computer descriptions to the diagnostic testing of the finished product. This dissertation, however, is restricted to the problem of automated printed circuit board design, which involves the placement of components on the board and the routing of wires to interconnect them.

Optimal solutions to the placement problem can be found only for boards containing no more than 20 components [5,6]. In the case of wire routing, optimal solutions can be obtained [20] for individual wires but not for an entire board. Modern printed circuit boards present large placement and wiring problems with up to 200 components being interconnected on one board. Therefore, hueristic procedures are used for finding reasonable, sub-optimal placements and wire routings for these boards. Comparing these hueristic procedures to determine the most effective techniques is extremely difficult, and to date, no realistic comparison of published algorithms has been made. Steinberg's 34 component problem [16] has been widely used for comparing placement algorithms [11,19], but it is too small to be considered realistic today. The only comparison of

wire routing algorithms is made in terms of the computer time required to route similar sized problems. One primary purpose of this dissertation is to begin a realistic comparison of hueristic design automation algorithms by publishing some large printed circuit board problems.

ILLIAC IV is a large parallel processing computer designed by the University of Illinois and built by Burroughs Corporation [1,2]. A unique situation therefore exists in which a public institution has access to the detailed design specification of a large modern computer. Such information is usually subject to industrial proprietary classification. Proper application of this information, particularly to the area of design automation should benefit the academic community as well as industry. The printed circuit board design problems found in the ILLIAC IV control unit are large, obviously realistic problems. Widespread use of these design specifications as test problems should produce significant new information about the placement and routing of printed circuit boards.

Chapter 5 proposes a new algorithm [27] for wire routing which is designed for interconnecting integrated circuit packages. The new routing algorithm combines the best properties of some existing techniques, reviewed in Section 5.1, to achieve faster speed and greater flexibility. The implementation of the basic algorithm is discussed in detail and results are presented for the wiring of ILLIAC IV sample problems. The wire routing program is also used to compare the results from various placement algorithms.

2. THE ILLIAC IV DESIGN PROBLEM

The Control Unit (CU) of the ILLIAC IV computer is composed of many large multilayer printed circuit boards. Each board can contain up to 165 integrated circuit packages arranged in eleven rows of fifteen packages each (Fig. 1). The integrated circuit packages employed are sixteen pins dual inline packages. The integrated circuits use high speed emitter coupled logic (ECL) technology with gate times in the range of two to four nanoseconds.

The placement and wiring of the ILLIAC IV CU boards represents a large collection of design automation problems. This set of problems was first solved by the Burroughs Corporation along with the University of Illinois. The high speed which was sought for the ILLIAC IV CU dictated a number of very strict wiring rules. First of all, a multilayer board was required so that ground planes would surround each signal layer in order to insure proper transmission line effects. The use of transmission line technology also restricted the manner in which connections could be made. Each connection was required to proceed from the source through each load until the final one where it was to be terminated by the proper resistance. Each pin to be connected could have only two wires connected to it so the connection had to be chained (Fig. 2). In addition, there was a minimum limit placed on the length of each connection in order to eliminate unwanted reflections. In certain special cases further restrictions were imposed. For the sake of timing the lengths of two wires were sometimes required to be identical which resulted in laying out portions of some boards completely by hand.

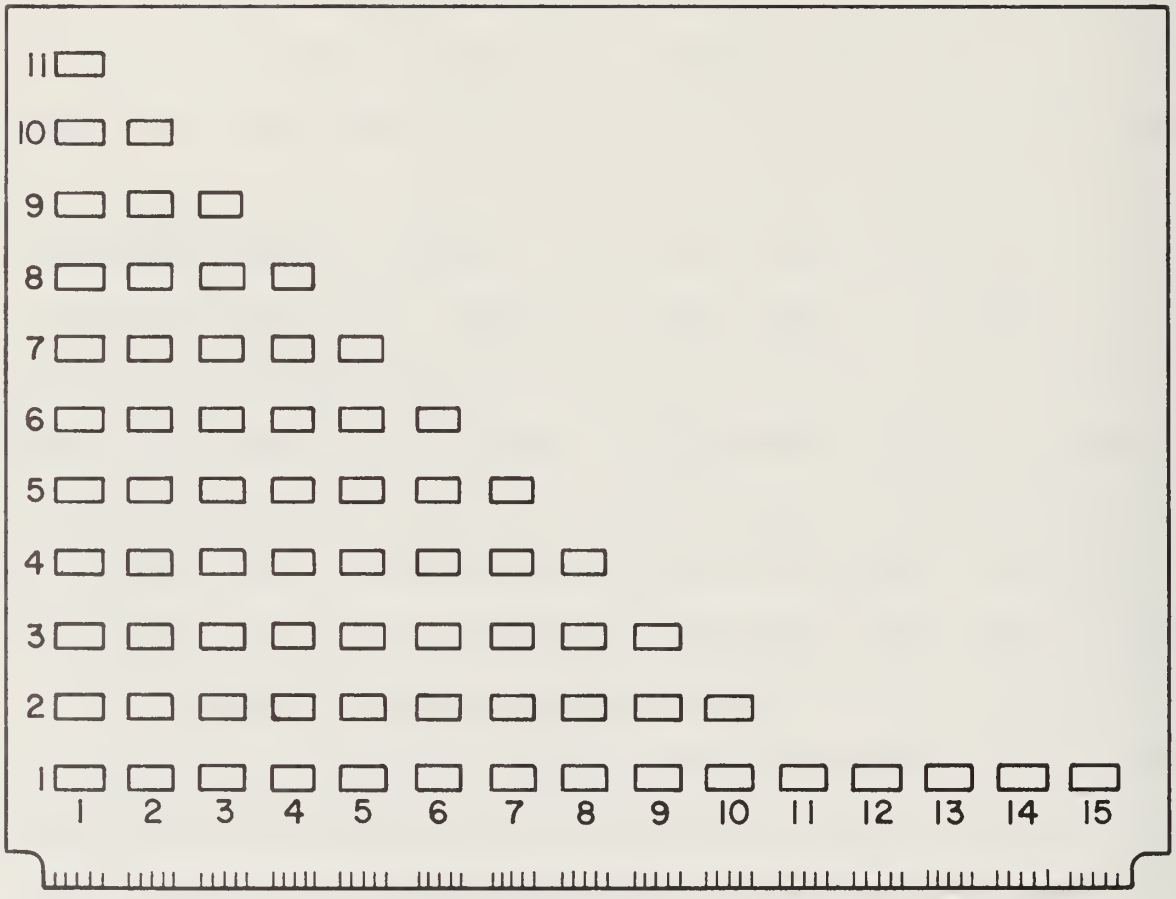


Figure 1. ILLIAC IV CU Board

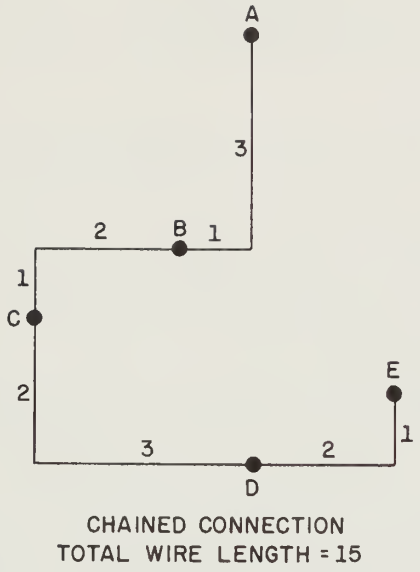


Figure 2. A Chained Connection.

For the purposes of this dissertation the CU board design problem will be somewhat simplified. What is desired is a set of large realistic design problems which are to be solved using a set of widely known and accepted wiring rules. The first step in simplifying the design problem is to consider only the signal connections to be made on each board. The specification of the wiring for each board is presented in the form of a netlist. By removing the power and ground connections from each netlist, specifications can be presented which determine only the signal connections to be made. Simplified netlists for five CU boards are listed in Appendix A. A second step in simplifying the design problems is to

specify a less restrictive set of wiring rules. The following set of rules was chosen to provide the greatest ease in comparing different placement and routing techniques:

1. No upper or lower limit is imposed on the length of any one connection.
2. No limit exists on the distance that two adjacent wires may run in parallel.
3. All wire segments are vertical or horizontal. (Manhattan geometry.)
4. No limit is placed on the number of wires which can meet at one point.
5. A two-sided board can be used for wiring.
6. Potential positions for packages as well as positions of edge connectors are fixed on the board.
7. Plated through communicating holes (Vias) can be specified by the design process and are not limited in number.

Such a simplified set of wiring rules is intended only to help in comparing the effectiveness and speed of a wide range of placement and routing algorithms. In order to be useful, design algorithms should be capable of handling more difficult problems.

The detailed specification of the CU board has also been simplified somewhat. The positions reserved for pulldown and terminating registers have been eliminated and the position of edge connectors has been slightly rearranged. The results of this dissertation are based on the following specifications:

1. Eleven rows of fifteen package positions (Fig. 1).
2. Package pins numbered from lower left in a counterclockwise fashion (Fig. 3).
3. Minimum wire spacing is 50 mils.
4. Packages are .75 inches apart in the vertical direction and .5 inches apart horizontally. (Fig. 4.) (The package spacing is retained from the original specification as an arbitrary reference.)
5. Edge connectors are arranged in fifteen groups of sixteen pins each. Each group is directly below a column of packages and the pins are numbered as shown in Figure 5. (Note that each group of edge pins can be treated as a package position for the purposes of placement and wiring.)

Once again the aim of this simplified specification is to make the CU board design problems easy to work with so that comparison of different algorithms can be facilitated.

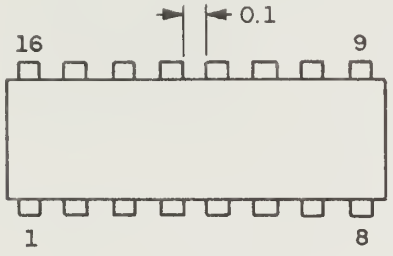


Figure 3. An Integrated Circuit Package.

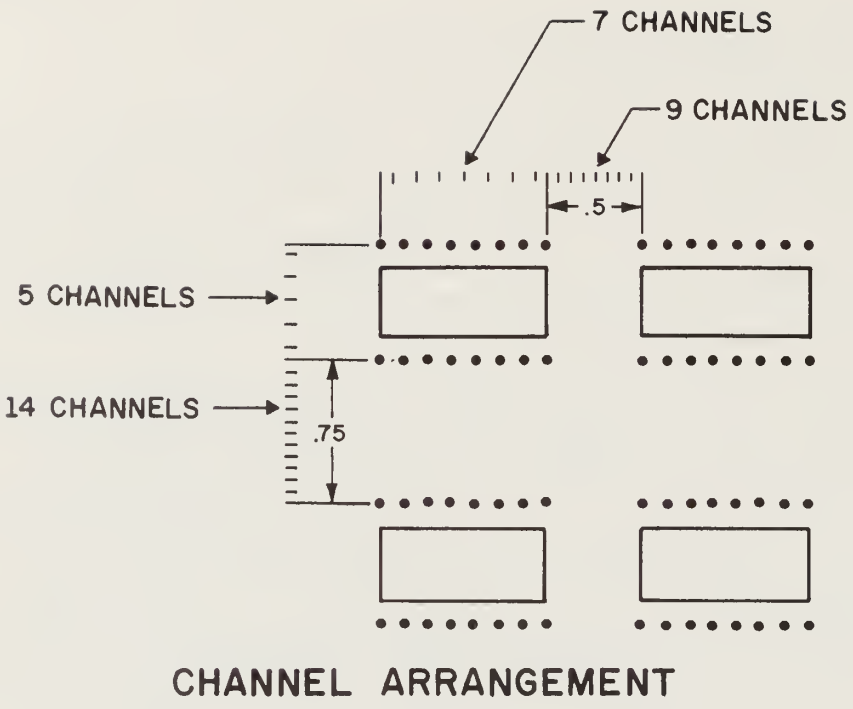


Figure 4. Package Spacing.

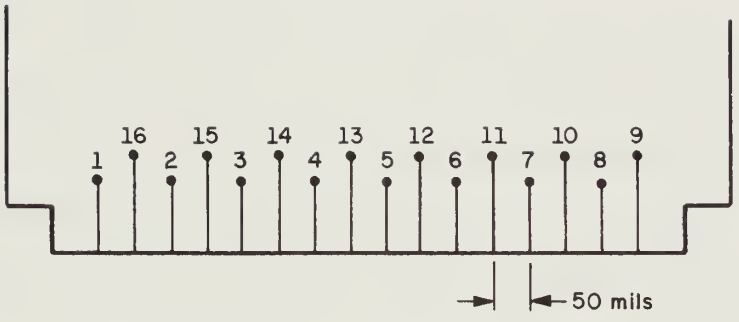


Figure 5. An Edge Connector Group.

3. GENERAL DESIGN AUTOMATION DISCUSSION

3.1 Measurement Techniques

The overall objective of a design automation system is to manufacture a piece of equipment with a given performance in the shortest possible time and for the lowest possible cost. Since decreasing the production time usually requires an increase in cost, some tradeoff must be made between these goals. The objective of each stage of the design process is also some minimization of time and cost. For the functions of placing and wiring printed circuit boards the amount of time can usually be measured directly, whereas the costs involved are quite complex. In order to minimize production costs it has been traditionally assumed that placement algorithms should strive toward minimizing the total wire length and the routing algorithm should strive toward minimizing the amount of area consumed in wiring. These measurements will be used throughout this paper to compare the effectiveness of various techniques. However, consideration will be given to the possibility of determining more meaningful measurements.

In order to minimize costs, a placement algorithm should produce a placement which can be wired efficiently under a given set of wiring rules. The ease of routing a board is typically dependent on the number of crossovers and the amount of congestion in a given area as well as on the total amount of wire that must be routed. It is generally assumed that decreasing the total wire length will decrease the number of crossovers and the amount of congestion in specific areas. This may

be true, but at the same time it is obvious that there are other ways of more directly improving the wireability of a board which, for instance, might take into account the amount of congestion in a given area. In Chapter 4 of this dissertation there is a quantitative evaluation of the factors contributing to the wireability of a board.

3.2 The Assignment Problem

The simple assignment problem can be demonstrated in terms of assigning modules to positions, where a specific cost can be determined for the assignment of a particular module to a particular position independent of the positions of other modules. If the objective is to minimize the cost, this can be accomplished by minimizing the sum of all individual costs, C_{ij}, of assigning module i to position j, where each module can be assigned to one and only one position. Munkres [4] has formulated a solution to this problem which is practical for up to 200 modules.

Placing a package on a printed circuit board, however, does not have a fixed cost (in wire-length) that is independent of the position of other packages. The placement problem cannot therefore be formalized as a simple assignment problem. The placement problem can be formalized as a quadratic assignment problem, but there are no known solutions to the quadratic assignment problem that are practical. The most obvious solution is to simply enumerate all possible assignments and determine which has the lowest cost (total wire-length). Such a solution requires n! assignments and cost evaluations and could be used only for the smallest

problems. Branch and bound techniques reduce the number of operations required and can be used for small n (15 - 20) as shown by Lawler [5] and Gilmore [6]. The computation time for these techniques increases exponentially as the number of packages increases.

As a result, the placement problem for large printed circuit boards has been approached by heuristic algorithms. Measurement of the comparative effectiveness of heuristic techniques is very difficult since optimum solution are usually unknown and since large realistic problems are not publicly available. Therefore, the thrust of the placement portion of this dissertation is the comparison of existing heuristic algorithms.

3.3 The Interconnection Problem

When several points on a printed circuit board are to be made electrically common, the interconnection which connects them is called a net. The manner in which a net is formed is often determined by the wiring rules of the board. The wiring rules adopted in Chapter 2 do not place any restrictions on the manner in which nets can be connected. Each net is therefore connected in such a way as to minimize the amount of wire used. The minimum spanning tree algorithm of Prim [7] is used to determine how each net is connected. This algorithm assumes that each connection must be made from one node on the net to another node on the net (Fig. 6a). If additional nodes are allowed to be added the result is a Steiner [9] tree which will always have a wire length less than or equal to that of the minimum spanning tree (Fig. 6b). However, Steiner trees are much more

difficult to find than minimum spanning trees [10] and they place considerable restrictions on the actual wire routing. Also, when Manhattan geometry is used, the difference between the wire length of a Steiner tree and that of the corresponding minimum spanning tree tends to be very small.

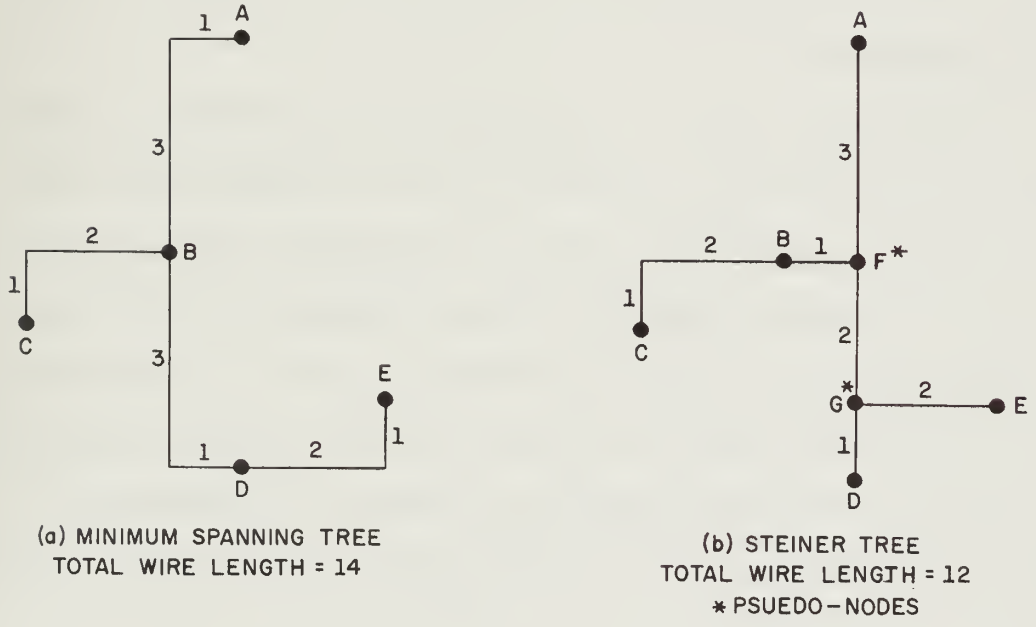


Figure 6. Minimum Spanning Tree Versus Steiner Tree.

4. PLACEMENT

4.1 Preliminaries

4.1.1 Connection Specification

In order to generate a placement it is necessary to obtain a complete specification of the interconnections to be made between components. This specification usually takes the form of a list of signal nets. The elements in this list contain the name of the component, the pin on that component to be connected, a source/load indication and the name of the signal net. When the list follows the format of each source being immediately followed by all of its loads the signal name is not needed to determine which nodes (list elements) are on the same net. This format will be used for specifying the signal nets for the ILLIAC IV CU boards (see Appendix A).

To facilitate the use of interconnection information a program was written to read the net list and generate cross reference tables containing all the needed information. All of the placement programs referred to in this paper work directly from these tables. The first table, COMPS, contains one entry for each pin of each component to be placed on the board. This table is stored as an array with one dimension being the component number and the other being the pin number on that component. Each entry in this table has one of four meanings, depending on its initial character.

1. The component and pin number to which this pin connects.
   Component names begin with an "A".
2. The edge connector pin of the board to which this connects.
   Edge connector pin names begin with a "P".
3. The number of the net to which this pin connects.
   Net numbers begin with a blank.
4. No connection is made to this pin.
   No connection is indicated by a "*".

4.1.2 Wire Length for One Package

Often times during the course of a placement algorithm it is useful to know the contribution made to the total wire length by the connections from only one package. Such information allows a computer program to determine whether moving one component to a new position will increase or decrease the total wire length. When such a component is connected to one or more nets it is very difficult to determine exactly how much wire is contributed by that component. The only way to find out how much wire is used is to measure the entire wire length of the net with the component being moved first in one position and then in the other. This measurement gives the change in the wire length of the net due to the moving of one component but it does not tell how much wire that component adds to the net in each position. The contribution made to the total wire length by the connections from only one package will not be measured. In each case the total wire length of the connections from one package will

be measured including the entire length of each net. Whereas such a measurement is not useful in itself, the difference in two of these measurements gives the exact change in the total wire length that results from moving one component to a new position.

4.2 Serial Placement Techniques

4.2.1 General Description

Serial placement algorithms assign packages to positions on a printed circuit board in only one pass. One package at a time is chosen and placed in a final positions, until all of the packages have been placed. This process can be broken down into two distinct subprocesses. The first is the selection process which chooses the next package to be placed and the second is the positioning process which assigns the chosen package to a position on the board. These two subprocesses are quite independent and several methods have been described for performing each of them [11]. The obvious advantage of using a serial placement algorithm is the speed of execution, with the number of operations performed being directly proportional to the number of units being placed. The obvious disadvantage of the serial techniques is that once a unit is placed, it can not be moved to compensate for further developments. Because of this rigidity in the serial methods, along with their speed, they are often used simply to generate initial placements for other placement algorithms.

The order in which units are to be placed is determined by maximizing some objective function. This function is dependent on the number of connections that each package has. One of the simplest such

functions is the pair linking method [14] which proceeds in the following manner. At any given time in the serial placement process, some packages have been placed on the board and some are available to be placed (Fig. 7). The pair linking method finds the package not already placed which has the greatest number of connections to one package which has been placed. Thus, the most strongly linked pair between the unplaced and placed packages has been found and the unplaced member of the pair is assigned to be the next package placed. At this time the selected package can be assigned to a final position on the board or it can simply be entered into the set of placed packages so that the entire order of placement can be determined. The process can be broken into two separate routines, the first one orders the packages and the second one positions them on the board. Different ordering and positioning algorithms can then be intermixed conveniently.

The cluster development [14] selection method uses a more complex objective function than the pair linking method. In cluster development, the next package to be placed is the one among the unplaced packages which has the greatest number of connections to already placed packages. Counting the number of connections to other units presents a slight problem for both the cluster development method and the pair linking method. The problem is to determine how many wires are needed to connect a package to a net. In Figure 6a, nodes A, B, C, D and E are all connected on one net. Whereas nodes, A, C and E are connected to the net by only one wire, node D is connected by two wires and node B is connected by three wires. At the time that one package is being placed, some of

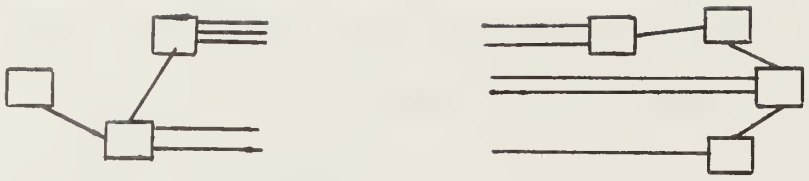


Figure 7. Package Selection.

the packages which are connected to it by nets may not be placed so that complete formation of the net is impossible. One solution to this problem is to form as much of the net as possible, for instance, to create a minimum spanning tree for the nodes which correspond to the already placed packages of the net. This approach will not be perfectly accurate so a less time consuming approximation should be acceptable. Such an alternative is to connect the package being placed only to the nearest node of the net in each case.

The motivation behind cluster development is to choose the package with the greatest number of connections to packages already fixed in position so that the chosen package can be placed with the greatest accuracy. A further criterion for choosing a package might be to minimize the number of new connections to unplaced units which it generates [15].

The idea here is to create the fewest possible restrictions on the future placement of packages. The realization of this idea is to choose the package which has the greatest difference between connections to already placed packages and connections to unplaced packages.

The second half of each serial placement algorithm is a routine which assigns each package to a final position on the board. The objective in placing each unit is to minimize the amount of wire that is added when connecting that unit. Only those wires which connect to already placed components may be considered. The most obvious and also the most time-consuming method for finding the best position for one package is to try it in every available position. The amount of wire used to connect the package at each position is measured, and the position which requires the lowest amount of wire is chosen as the final position. In order to decrease the amount of time involved several simplifications can be made.

Rather than trying each unit in every available position a subset consisting of the most likely positions can be tried. Since the edge connectors on an ILLIAC IV CU board are all on one edge (Fig. 1), the placement of packages will begin at that edge and grow away from it. In this case a subset consisting of the first available position in each column of package positions (open boxes in Fig. 8) could be used. This selection limits the number of positions tried for each package to fifteen. A much more flexible set of candidates can be conveniently generated involving a slightly larger subset. This larger subset is made up of the positions which "outline" the previously placed packages (all boxes in

Fig. 8). Although it can be shown that the optimum position for a package may not be contained in the "outline," this method is much faster than trying all available positions.

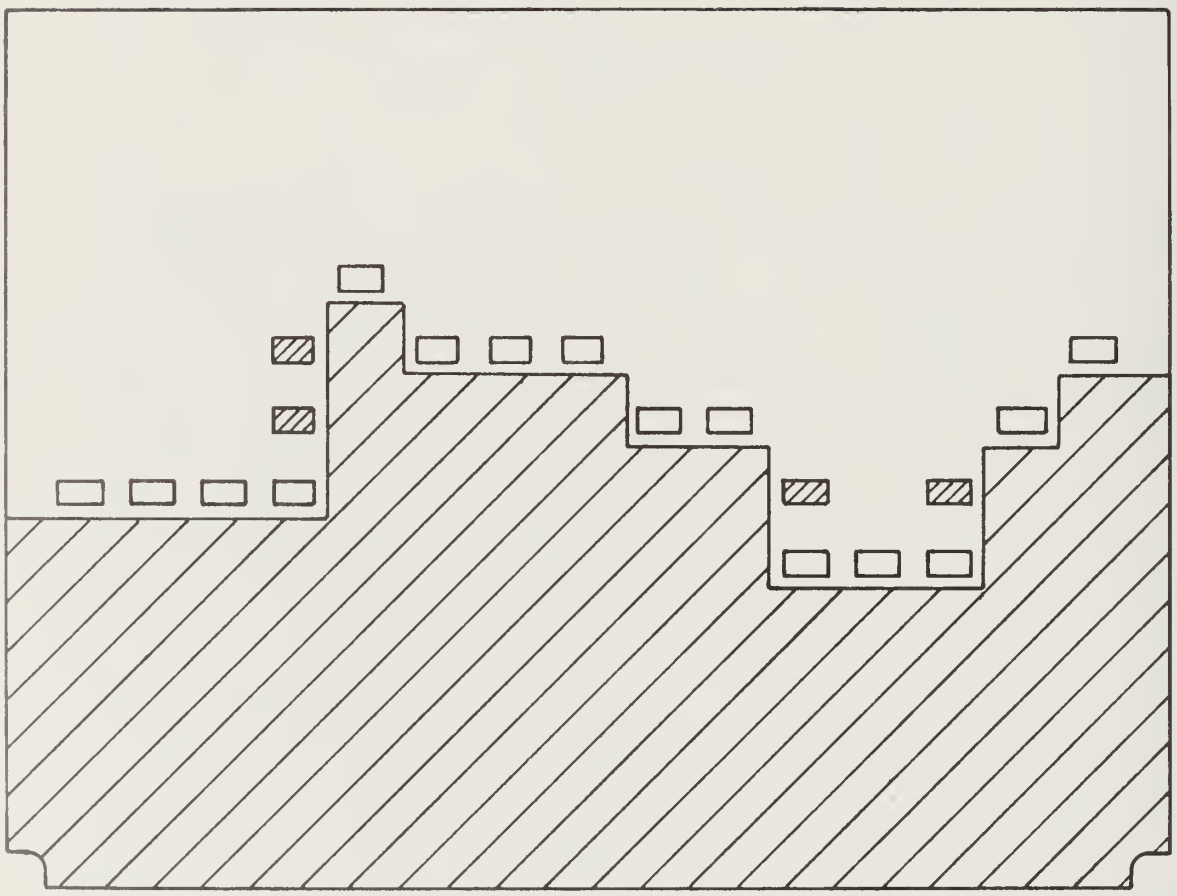


Figure 8. Candidate Positions.

4.2.2 The Serial Placement Program

One serial method was programmed and used to produce placements which could be compared to those generated by other techniques. The selection and positioning rules employed in the program were chosen mainly for their ease of implementation and speed of execution. No attempt was made to find the "best" serial placement method. The intention here was to implement a reasonable technique to be used as the basis for some meaningful comparisons. The placement method which is described in detail below, will be referred to simply as the Serial Placement method for the remainder of this dissertation.

The selection rule employed chooses the package with the greatest number of connections to the already placed packages. Initially, the only packages placed on the board are the edge connectors which interface with the backplane. Therefore, the first package chosen for placement is the one which has the greatest number of connections to connector pins. Implementing such a scheme is quite straightforward.

Step 1. Initialize c_i. Each package is assigned a value c_i which indicates how many connections it has to connector pins.

Step 2. Select one package for positioning. Find the maximum c_i and select the corresponding package.

Step 3. Update c_i. For each package connected to the most recently selected package $c_i = c_i + n$ where n is the number of times it is connected.

Step 4. If all packages have not been selected yet, return to Step 2.

In the algorithm above, Step 3 is very important. The most recently chosen package may be connected to a net (a connection involving more than two packages). When this occurs, each package on the net will have its c value increased by one. Each package on a net is therefore considered to be directly connected to every other package on that net rather than just being connected to the net.

Each time a new package is indicated as the next to be placed it is tried in several of the available positions to determine which one will require the least amount of wire to connect it. In order to save time, the new package is not experimentally connected at every available board position. Only those positions adjacent to already placed packages or to board connector pins are investigated as depicted in Figure 8. The measurement which is made at each position is simply the Manhattan distance for each point-to-point connection plus the distance to the nearest node of a net. In many cases the package to which a connection should be made has not yet been placed. In such instances the connection is completely ignored, and for this reason the proper formation of nets could not be employed, so the above simplification was used.

4.2.3 Serial Placement Results

The Serial Placement program was used to place 20 different ILLIAC IV CU boards (Table 1). These 20 boards constitute a representattive subset of the entire set of 64 different CU boards. This subset was chosen on the basis of total wire length and number of packages. The program was run on a Burroughs 6500 computer which has a five megahertz basic

Table 1. Serial Placement Results

Board Name	Number of Packages	Total Wire Length	Placement Time (sec)	Vertical Overflow	Horizontal Overflow
FIRDB	52	725	19	0	1
FDS	63	1102	18	0	0
IWSCTL	83	1159	24	0	0
FDQA	93	1230	25	0	0
FIRCB	102	1292	27	0	0
FORC	96	1313	28	0	0
IIAA	118	1340	34	0	0
FWLDF	102	1346	22	0	0
ATP07	101	1429	27	0	4
MTMCTL	114	1460	25	0	0
MPCTL	99	1527	29	0	0
TXFER	97	1608	22	0	5
ICTLA	110	1637	26	0	0
MDSPLY	75	1736	20	16	32
ATP05	112	1851	46	43	0
FBUSY	119	2058	28	24	3
ATP25	134	2126	41	0	0
TCRFLD	136	2318	30	16	9
ILTCL	141	2416	37	37	0
IICR1	121	3306	38	42	78

clock speed. The placement times shown in Table 1 give the processor time used. The Channel Routing program described in Chapter 5 was used to route the boards placed by the Serial Placement Program. The routing results are intended to measure the wireability of the placements and to determine whether wire length measurement can be used to predict wireability.

The last two columns of Table 1 indicate the success of wiring the boards when the original package spacing for ILLIAC IV is used. This spacing shown in Figure 4 allows a total of 19 channels for each package row and 16 channels for each column. The edge connector pins are located 1.7 inches from the first row of packages allowing 33 horizontal channels in that area. The number of overflows in Table 1 is the number of additional channels needed after final positioning to completely wire the board. In Table 2 additional routing information is given. The last two columns of Table 2 indicate the minimum number of channels which must be available in each row or column of packages for complete wiring using the Channel Router. The wireability measure is intended to indicate the total amount of successful wiring as well as the distribution of wiring density over the board.

Table 3 is the correlation matrix of several of the variables present in Tables 1 and 2. All of these variables are shown to be related by the fact that all correlations are positive with most of them above .5. A few of these results are considered significant. The high correlation (.88) between the number of connector pins used and the number of vertical channels required may be helpful in partitioning logic onto boards. Such

Table 2. Wire Routing of Serial Placements

Board Name	Connector Pins Used	Total No. of Vertical Channels	Total No. of Horizontal Channels	Vertical Wireability	Horizontal Wireability
FIRDB	57	121	116	13	20
FDS	156	201	112	16	18
IWSCTL	173	181	128	15	17
FDQA	91	171	148	13	17
FIRCB	146	185	142	13	15
FORC	84	172	146	16	18
IIAA	90	163	169	15	17
FWLDF	149	187	131	14	15
ATP07	156	207	137	15	21
MTMCTL	135	176	137	14	15
MPCTL	154	195	158	16	18
TXFER	224	222	145	16	21
ICTLA	176	221	165	16	18
MDSPLY	223	257	154	20	26
ATP05	178	250	214	30	19
FBUSY	220	265	184	18	21
ATP25	170	232	215	16	18
TCRFLD	196	230	205	19	22
ILTCL	211	278	192	22	18
IICR1	221	283	303	21	32

Table 3. Correlations

		1	2	3	4	5	6	7
1.	Number of Packages	1.00	.40	.73	.56	.67	.36	.04
2.	Connector Pins Used	.40	1.00	.69	.88	.48	.53	.50
3.	Total Wire Length	.73	.69	1.00	.86	.93	.59	.66
4.	Total No. of Vertical Channels	.56	.88	.86	1.00	.71	.72	.89
5.	Total No. of Horizontal Channels	.67	.48	.93	.71	1.00	.62	.64
6.	Vertical Wireability	.36	.53	.59	.72	.62	1.00	.42
7.	Horizontal Wireability	.04	.50	.66	.89	.64	.42	1.00

a measure should help predict the difficulty of wiring the boards in a given partition. The total wire length correlates highly with the number of vertical (.86) and horizontal (.93) channels required. However, total wire length does not predict wireability consistently.

4.3 Iterative Placement Techniques

4.3.1 General Description

All iterative placement techniques begin with a complete placement of a board. These initial placements can be generated manually or by random selection or by another placement algorithm. Each step in the iterative technique changes the positions of a subset of the packages so that the total wire length of the board is decreased. Thus, after any iteration, the board is completely placed and the process may be terminated. The algorithm continues until some stopping criterion is met. Most times these methods are stopped when no further decrease in the wire length is being made. The advantage of iterative schemes is that they produce better placements than the serial methods. Even though some iterative techniques are very simple to program they all tend to require large amounts of processing time.

An exhaustive pairwise interchange is the simplest iterative scheme. Each iteration consists of interchanging two packages on the board and determining whether that interchange results in a decrease in the total wire length. If no change occurs or if the total wire length is increased, the packages are returned to their previous positions. An exhaustive pairwise interchange systematically attempts every possible interchange between two packages on the board. The exhaustive pairwise

interchange can be repeated until none of the interchanges result in a decrease in the total wire length. A local minimum has then been reached. An exhaustive three-way interchange is a similar technique which tries all possible interchanges of three units in each iteration. It has been shown however, that the results of the three-way interchange are not significantly better than those of the pairwise interchange [29].

Steinberg's algorithm [16] considers a large subset of all the packages during each iteration using an optimal procedure for repositioning that subset. The basic subset used in Steinberg's algorithm is an unconnected set of packages. An unconnected set (Fig. 9) is made up of packages, such that no package in the set is connected to any other package in the set. Therefore, the positioning of any one package of an unconnected set is completely independent of the positioning of the other

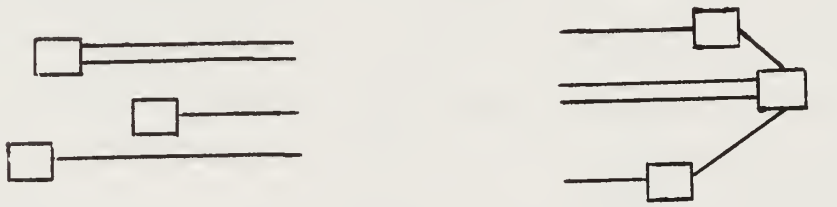


Figure 9. An Unconnected Set.

packages of the set. The placement of an unconnected set of units in the available board positions (Fig. 10) created by their removal is therefore a simple assignment problem for which an optimal solution can be easily found [4]. The positioning of the unconnected set of units will always either decrease or not change the total wire length of the board.

There are many unconnected sets that can be formed from any given set of packages. The process for constructing an unconnected set starts by choosing a package at random and proceeds by adding to the set only those packages which are not connected to members of the set. This process is continued until a set of the desired size has been formed or until no more packages can be added to the set. An unconnected set which can not be added to, is known as a maximal unconnected set. The size of the unconnected set to be used will be largely determined by the computing power available for solving the assignment problem. It is always preferable to use maximal unconnected sets when possible to ensure the greatest flexibility in the assignment.

Rutman [17] suggested an improvement to Steinberg's algorithm to produce better placements and to speed up the convergence of the algorithm. After some predetermined number of Steinberg iterations Rutman inserts an interchange step which decreases the length of the longest wires. Then Steinberg's algorithm is applied for a few more iterations. Rutman's interchange seems to help keep Steinberg's algorithm from stopping at a local minimum.

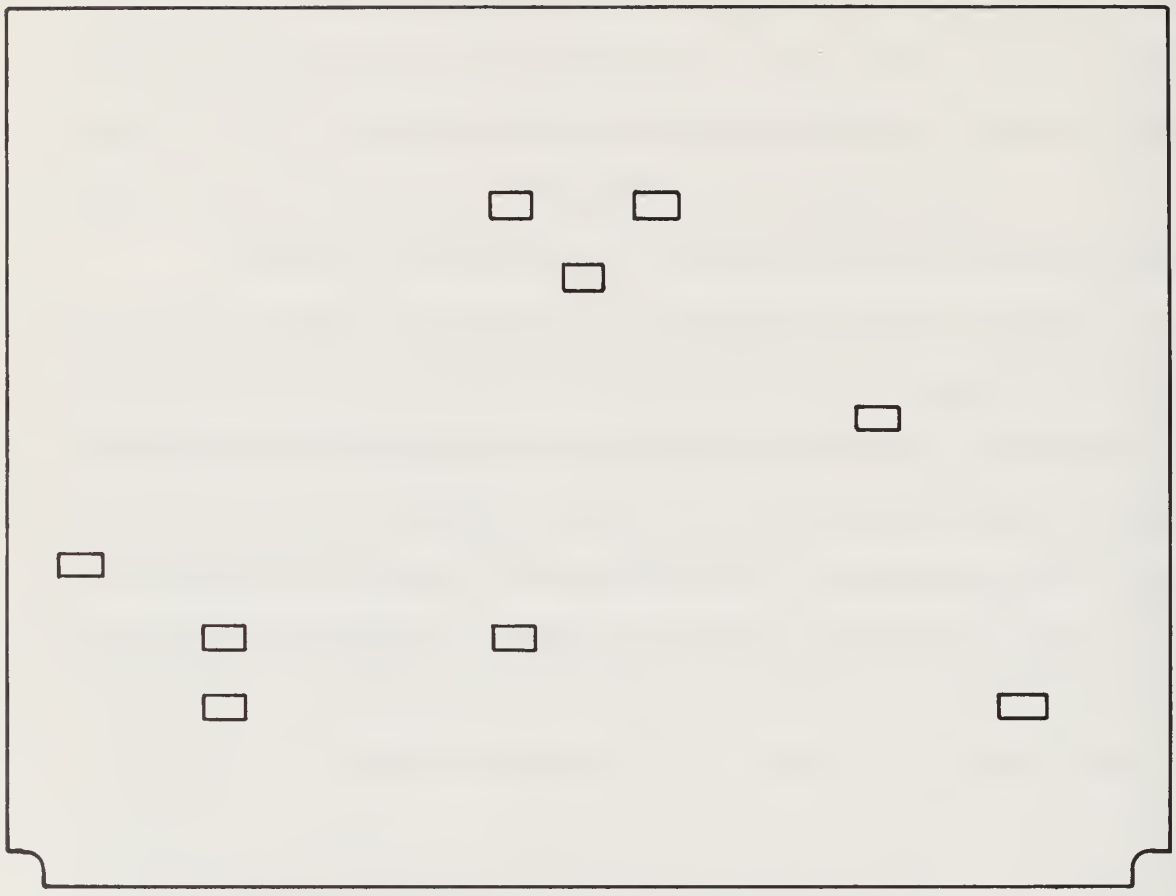


Figure 10. Board Positions of an Unconnected Set.

Another iterative placement method employs a relaxation principle. After an initial placement is established a measurement is made for each package which determines how strongly it is connected to other packages. This measurement represents the degree to which one package wants to be close to another. Each of these measurements can in turn be represented by a force which acts to move a package closer to the packages it connects to. An equilibrium position which balances the forces on each package is then computed for all packages. The new placement is then formed by moving all packages to the position closest to their equilibrium positions with the constraint that no two packages may occupy the same position. Other relaxation methods are quite similar to this description and are beginning to appear in the literature [11]. One difficultly in implementing a relaxation algorithm is the problem of representing net connection so that they realistically influence positioning during each iteration.

4.3.2 The Pairwise Interchange Program

The pairwise interchange program exhaustively attempts all possible interchanges of two packages. The change in the total wire length of the board that results from each interchange is determined, and only those interchanges which decrease the total wire length are actually made. If any interchanges are performed during this process then the board positions will be different at the end than they were to begin with. Further improvement may be possible by simply trying all possible interchanges on this new board placement. Continued

iteration through all possible pairwise interchanges may or may not continue to improve the total wire length of the board. As soon as one iteration fails to produce an improvement a local optimum has been reached which has the property that no interchange of any two packages will result in a reduced total wire length for the board. This local optimum is not unique. Its configuration depends on the initial placement of the board.

Implementing the Pairwise Interchange method is quite straight forward; however, some care must be employed to keep the execution time from becoming unreasonable. When considering the interchange of two packages, it is not practical to measure the entire wire length of the board before and after the interchange. Only the change in the total wire length is important. If all of the connections to the two packages in question are point-to-point connections this calculation becomes trivial. The difference in total wire length is simply the change of length of each connection. Frequently, however, a package will be connected to a net which is a single connection of more than two points on the board. The amount of wire needed to connect a package to a net can not be determined without reconstructing the entire net each time this information is needed. Since the structure of the net may be changed significantly by moving one package the change in total wire length will only be correct if it is determined by taking the difference of the entire length of each net for the two package positions. The difference

in total wire length measured in this manner will be the same as when the total wire length is computed for each position and then the difference taken. The results will therefore be the same while a great increase in speed will be achieved.

The algorithmic description of the pairwise interchange program is given below, where P(I) represents the I^{th} package, and N is the number of packages.

Step 1. Initialize. Set initial positions of packages on the board. Set I to 1 and J to 2. Set IND to 0.

Step 2. Measure the wire length of the connections to P(I) and P(J).

Step 3. Interchange P(I) and P(J).

Step 4. Measure the wire length of the connections to P(I) and P(J) in their new positions and determine the total change in wire length (ΔW).

Step 5. If ΔW is positive, interchange P(I) and P(J) returning them to their original positions. Otherwise set IND to 1.

Step 6. If I = N go to Step 7. Otherwise if J = N set J to 1 and I to I + 1 and go to Step 2. Otherwise set J to J + 1 and go to Step 2.

Step 7. If IND = 0 then done. Otherwise go to Step 1.

4.3.3 Pairwise Interchange Results

Since the iterative placement programs required large amounts of computer time, a set of only five boards was chosen to compare them. The complete net lists for these five boards are given in Appendix A. Each of the iterative placements used the Serial Placement results as initial placements. Also, the iterative placement interchanges were restricted to the board positions filled by the Serial Placement program.

Each iteration of the Pairwise Interchange program attempts all possible pairwise interchanges of packages on the board. These iterations are repeated until one produces no improvement in the wire length indicating a locally optimal solution. The Pairwise Interchange Results in Table 4 include this final iteration. The execution time for each iteration of a given board placement remains about the same, decreasing slightly as the number of interchanges decrease. The execution times given in Table 4 are the processor times for execution on a Burroughs 6500 computer and can be directly compared to the Serial Placement times.

4.3.4 Implementation of Steinberg's Algorithm

Steinberg's placement algorithm [16] was the most difficult of the three methods to implement; therefore, it was implemented in its most simplified form. The algorithm consists first of finding a set of packages such that a member of the set does not connect to any other member of the set. Since the members of such a set are independent, they can be interchanged by a simple procedure which will insure that the total wire length of the board will be improved or at worst stay the same. Repeated application of this procedure to different sets of packages will continue to decrease the total wire length until some local optimum is reached.

Table 4. Pairwise Interchange Results

Board Name	No. of Iterations	No. of Inter-Changes	Execution Time (Min.)	Starting Wire Length	Final Wire Length	Improve-ment
FIRDB	2	14	24	725	700	3.5%
FDQA	4	57	84	1230	1085	12 %
ATP07	3	45	55	1429	1320	7.5%
TCRFLD	3	41	67	2318	2243	3 %
IICR1	4	148	136	3306	2845	14 %

A set of packages which have no connections to other packages within the set is called an unconnected set and is illustrated in Figure 9. Such a set can be easily formed in the following manner:

Step 1. Initialize. Mark all packages to be unconnected.

Step 2. Select one package at random. If it is marked connected repeat Step 2.

Step 3. Mark the selected package and all packages it connects to, as connected. Enter the selected component in the unconnected set.

Step 4. When the required number of packages are in the unconnected set or when all of the packages have been marked unconnected, stop. Otherwise return to Step 2.

When an unconnected set is formed such that no other package can be added to the set, it is called a maximal unconnected set. The above description represents the implementation which was employed; therefore, the unconnected sets are formed in a random manner. The size of the unconnected sets is kept as large as possible and often times maximal unconnected sets are used.

Since the elements of an unconnected set are by definition not directly connected to one another, the positioning of each member of an unconnected set is completely independent of the positioning of the other members. Therefore, the problem of optimally placing an unconnected set of packages with respect to minimum wire length reduces to a simple linear assignement problem which can be quickly solved by Munkres Algorithm [4].

There is one small problem which must be handled properly. In setting up the assignment problem a measurement must be made of the amount of wire needed to connect a package in each of the available positions. As was the case in the Pairwise Interchange method, each time a net is encountered during this measurement, the entire net must be reconstructed and the entire net wire length must be added. Each entry, C_{ij}, of matrix C is therefore the wire length used in connecting package i in position j. The difference, $c_{ij} - c_{ik}$, gives the exact change in wire length for moving package i from position j to position k.

The following algorithmic description of Munkres method is given by Rutman [17] in his implementation of Steinberg's algorithm.

Preliminaries

In the matrix [C], no lines are covered and no zeros are starred or primed. Consider row 1 of the matrix. Find the smallest element in row 1, and call it h. Subtract h from each element of row 1. In the process of subtracting h, some element (or elements) of row 1 become zero. Whenever an element becomes zero and if there is no starred zero in the row and none in its column, star the zero and cover the column containing the zero. If there is no zero star in its column but there is one in its row, add the matrix coordinates of the zero to the zero

A list (unless the zero A list is already full). Do the same for each row of the matrix.* If N columns are covered, the starred zeros form the desired result, and the problem is finished. If less than N columns are covered, go to Step 1.

Step 1

Take the coordinates of a zero from the zero list, popping up the zero A list. If the zero is covered, push the coordinates of the zero into the top of the zero B list and take another zero from the zero A list. Continue until a non-covered zero is found or until the zero A list is depleted. If the zero A list is depleted and flag F is set search the matrix for an uncovered zero. If the zero A list is depleted and flag F is reset, go to Step 3.

Assume an uncovered zero is found. Prime the zero, then consider the row containing it. If there is no starred zero in this row, go at once to Step 2. If there is a starred zero in the row, cover the row and uncover the column of the starred zero. Look for any uncovered zeros in this new uncovered column. If any exist, add their coordinates to the top of the zero A list (if the zero A list overflows, set flag F). Then go to beginning of Step 1.

---

* Note that at this time the zero A list (if sufficiently large) will contain the coordinates of every uncovered zero if any exist. If there are too many zeros for the zero list to hold, flag F will be set. Otherwise, flag F is reset.

Step 2

There is a sequence of alternating starred and primed zeros, constructed as follows: Let Z_0 denote the uncovered 0 (there is only one). Let Z_1^* denote the 0* in Z_0^1's column (if any). Let Z_2^1 denote the 0 in Z_1^*'s row (if Z_1^* exists, its column is not covered since Z_0^1 is in its column and Z_0 is not covered, so its row must be covered. Hence there is a 0 in this row (see Step 1). Let Z_3 denote the 0^* in Z_2's column (if any). Similarly, continue until the sequence stops at a 0 which has no 0^* in its column (Munkres proves such a 0 exists and the sequence is unique)).

Now unstar each starred zero of the sequence, and star each primed zero of the sequence. Erase all primes (that is, remove the zero prime coordinates from vector C_p). Uncover every row and cover every column containing a 0^*. If N columns are covered, the starred zeros form the desired result and the algorithm is finished. Otherwise, empty the zero B list into the bottom of the zero A list and go to Step 1.

Step 3.

Let h denote the smallest uncovered element of the matrix; it will be positive. Add h to each twice-covered element. Subtract h from each uncovered element. In the process of subtracting h, add the coordinates of any new zeros to the zero A list. Return to Step 1 without altering any stars, primes, or covered lines.

4.3.5 Steinberg's Placement Results

The Steinberg Placement program was tested on the same problems as the Pairwise Placement program (4.3.3). The results of the Steinberg placements are presented in Table 5. Each iteration involves choosing and repositioning one unconnected set. The program was stopped when 50 consecutive iterations produced a decrease in wire length of less than 10 units. For each placement except that of board IICR1 it appeared that no further improvement would be made in the wire length. The placement of IICR1 had improved by 15 units in each of the last two sets of 25 iterations and would probably continue to improve slightly. It was terminated after one hour of processor time for monetary reasons. The Steinberg Placement program was run on a Burroughs 6700 computer which is a faster modified version of a 6500. The execution times for Steinberg results can not be directly compared to other placement execution times. The basic clock speed of the 6500 and 6700 are the same but the 6700 is faster on some operations. The conclusion indicated is that the Steinberg Placement program is faster than the Pairwise Placement program, but not twice as fast.

Table 5. Steinberg's Algorithm Results

Board Name	No. of Iterations	Size of Unconnected Set	No. of Inter-Changes	B6700 Execution Time (Min.)	Starting Wire Length	Final Wire Length	Improve-ment
FIRDB	100	Max (8-12)	4	8	725	702	3 %
FDQA	150	Max (15-20)	23	29	1230	1122	9 %
ATP07	150	Max (25-30)	32	49	1429	1320	7.5%
TCRFLD	150	30	32	33	2318	2181	5.5%
IICR1	150	Max (25-30)	75	62	3306	2733	17 %

4.4 Comparison of Results

The results of applying the Channel Routing program to the Pairwise and Steinberg placements are presented in Table 6. These results are used to measure the effectiveness of applying iterative placement algorithms to serial initial placements. The only clear conclusion is that the change in wire length does not effectively predict the change in wireability as defined in Section 4.2.3. Both the Pairwise and Steinberg program produce improvements in wireability in some cases but in most cases no change in wireability occurs. Generally, not enough improvement is made to justify the computer time involved in the iterative techniques used in this study.

Table 6. Comparison of Results

Board Name	Placement Method	Total Wire Length	Vertical Overflow	Horizontal Overflow	Vertical Wireability	Horizontal Wireability
FIRDB	Serial	725	0	1	13	20
	Pairwise	700	0	0	11	16
	Steinberg	702	0	2	12	20
FDQA	Serial	1230	0	0	13	17
	Pairwise	1085	0	0	12	17
	Steinberg	1122	0	0	12	17
ATP07	Serial	1429	0	4	15	21
	Pairwise	1320	1	2	17	20
	Steinberg	1320	0	1	15	20
TCRFLD	Serial	2318	16	9	19	22
	Pairwise	2243	25	6	20	21
	Steinberg	2181	18	9	19	22
IICR1	Serial	3306	42	78	21	32
	Pairwise	2845	33	38	19	27
	Steinberg	2733	12	34	17	27

5. ROUTING

5.1 Review

5.1.1 Lee's Algorithm

Lee's algorithm [20] is an exhaustive search technique which always finds the minimum distance path between two points on a plane, if such a path exists. The plane is first divided into a very fine grid where the length of each side of a grid square is equal to the minimum distance between parallel running lines on the plane (Fig. 11). The grid square containing one of the two points to be connected is labeled the origin (O) while the grid square containing the other point is labeled the destination (D). The path finding algorithm begins by labeling the four grid squares immediately adjacent to the origin with the number 1 if they are available to contain part of a path. Next, all of the available squares that are immediately adjacent to squares containing 1's are labeled with the number 2. In general, at any step K, all of the available squares which are immediately adjacent to squares labeled K-1 are assigned the label K. This procedure continues until the destination square is encountered at which time the path can be constructed by following the reverse sequence of numbers from the destination square back to the origin. Several minimum paths may exist between the two points to be connected, so some additional rule must be chosen to determine which path will be used. Such a rule might simply be to always travel as far as possible in one direction before making a turn.

There are two major drawbacks to Lee's algorithm. First of all, the number of grid squares that must be considered is very large since each one is so small. This fact causes the amount of storage needed to

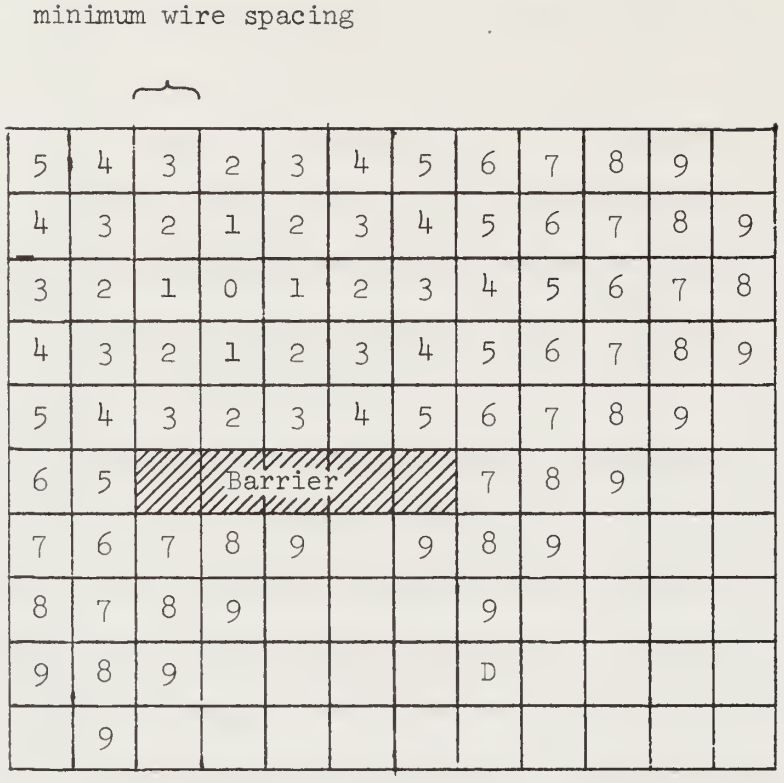


Figure 11. Lee's Algorithm.

implement the algorithm to be very large. This storage requirement can be greatly reduced by using a simplified labeling sequence as suggested by Ackers [25]. Rather than recording the distance of each grid square from the origin, a pattern of two 1's following by two 2's is generated emanating from the origin. A minimum path can be constructed by retracing this pattern from the destination back to the origin and its length can be determined while retracing. The storage requirement for each grid square will thus be reduced to two bits. Even with this improvement the storage requirement is large. The second major drawback of Lee's algorithm is the amount of computation time required. The number of operations needed to find a path of length n is proportional to n^2 which is responsible for the fact that the amount of computation time used is unreasonable for large wire routing problems.

The main advantage of Lee's algorithm is that it will always find the minimum path between two points, if any possible path exists. Such an algorithm is most useful when only a few remaining connections need to be made on an almost wired board. The problem with wiring an entire board using Lee's algorithm is that each connection is finalized as soon as it is found and these finalized connections may interfere with future attempts to construct connections. This interference may only cause a slight increase in the length of a future path, but it may also unnecessarily eliminate the existence of any path for some connections. It has been found that these adverse effects can be minimized by always connecting the shortest wires first and then the longer ones. In any event, only one connection is considered at a time so that optimizing the wiring of an entire board should not be expected.

5.1.2 Line Routing

One alternative to Lee's algorithm is to create a series of line segments rather than looking at each grid square individually. Methods which use this approach will be referred to as line routing techniques. One line routing algorithm was developed by Hightower [21] which is briefly described in the following explanation.

The path is constructed from both the origin and destination points simultaneously. From the origin a verticle line segment is constructed which extends in both directions as far as possible without intersecting any finalized connection or obstacle on the board. A similar line segment is constructed horizontally at the destination point. Intersection between any line segment connected to the origin and any line segment connected to the destination indicates that a path has been found and the procedure can be terminated. If no intersection is found an escape line is constructed from each of the line segments generated by the previous iteration. An escape line is a line perpendicular and intersecting a previous escape line which extends the horizontal or vertical boundaries established by the set of previous escape lines. This procedure of finding escape lines is repeated until a path is found or until no further escape lines can be found, indicating that no path exists.

While the line routing algorithm of Hightower does not insure that the minimum path will always be found, it does have distinct advantages over Lee's algorithm. The amount of storage space required is reduced because only information about existing line segments must be maintained. Also, the number of operations performed is greatly reduced so

that significant time savings are realized. As with Lee's algorithm, Hightower's line routing algorithm considers only one connection at a time so that wire length of succeeding connections may be greatly affected.

5.1.3 Cellular Wiring

Wire routing using the cellular modeling technique has been introduced by Hitchcock [23]. This method of wire routing can be viewed as a modification of Lee's algorithm where the grid squares are enlarged so that several wires can pass through one cell (the enlarged grid square) without interfering with each other. The size of these cells is optional and Hitchcock chose to have the size and shape depend on the physical location of package pins on the board. Each cell has four edges through which wires can pass the capacity of each edge is determined by the number of wires which can fit through when placed at minimum spacing. Starting from the cell containing the origin a list of reachable cells is obtained which is expanded in a similar manner to Lee's technique. This process is repeated until the destination is encountered at which time the path can be constructed.

The internal structure of each cell need not be finalized until after all connections have been made. The fact that a wire passes through a cell can be noted by simply indicating the edge that the wire enters and the edge which that wire leaves. A cell is determined to be reachable if a wire segment in an adjoining cell can reach the edge between them and if the capacity of that edge has not already been filled. Once a wire segment enters one edge of a cell it is possible to determine which edges

are available for that wire segment to leave. The availability of an edge depends on whether the capacity of that edge has been filled and also on the topology of the wire segments passing through the cell. For example, if a wire segment already passes through the east and west edges of a cell, then the north edge will not be available to a wire segment entering from the south.

The cellular wire routing technique does not insure minimum wire length for each path; however, it does produce improvements over Lee's algorithm by reducing the amount of storage space required and by significantly reducing the computation time. Since the search procedure is the same as Lee's, the number of operations needed to find a path of length n (where n is now the number of cells) is still n^2. Although n will be smaller for the cellular approach since the grid size is larger, the increase in operations will again be quadratic as increasing board sizes are considered. Hitchcock's approach does have one distinct advantage over Lee's and Hightower's algorithms in that none of the connections are completely finalized until all of the connections have been formed, so that a connection may have fewer detrimental effects on the possibility of finding succeeding connections and on the amount of wire used in constructing succeeding connections.

5.1.4 Two-Layer Routing

The previously discussed routing algorithms have been explained in terms of making connections on a single layer of a printed circuit board. Lee's algorithm 5.1.1 has been extended to a two-layer wiring environment by Heiss [24]. The technique is simply to extend the search

pattern on both layers of the board simultaneously, and add a predetermined distance penalty for changing layers. This method has all the time and storage difficulties of Lee's algorithm, only doubled. A similar extension can be applied to cellular routing 5.1.3 in order to provide two-layer capability. The extension suggested by Hightower [21] for applying line routing to a two-layer board is the obvious solution which is to fill up one layer without crossovers and then begin on a new layer to attempt the remaining connections. The problem with each of these extensions is that they do not take full advantage of the possibilities of two-layer routing.

The line routing algorithm explained by Mikami and Yabucki [22] uses an attack similar to that of Hightower with the exception that all vertical wire segments are located on one side of a two-layer board and all horizontal wire segments are located on the opposite side. Via holes are therefore required to correspond to the end points of each wire segment. Hightower's routing program is capable of producing similar results by running in a "crossover" mode with the resulting wire segments being assigned to horizontal and vertical layers with appropriate vias. In fact, although he does not say so, it appears that this is exactly the method he used to produce his "Two Sided Board" example.

Stanley Lass introduced a method using a stepping aperture [26] which was specifically designed for two layer routing in an environment containing a fixed array of vias. Lass first wires all connections on the board without regard for conflicts between connections. A small aperture is then passed over the board with finalized wiring being

performed only within the current boundaries of the aperture. If any wiring can not be completed within the aperture, an attempt is made to shift the connection concerned so that it may be completed within a succeeding aperture position. Connections can generally be shifted only to the nearest aperture positions without greatly increasing their wire length. Therefore, wiring failures will occur when one area of the board becomes very crowded as well as when the entire board area becomes filled. Lass's technique embodies the advantages in time and storage provided by line routing, and has the added flexibility of allowing connections to be readjusted after their initial placement.

5.2 Channel Routing

5.2.1 Objectives

The channel routing algorithm proceeds in a manner quite similar to the stepping aperture approach of Lass 5.1.4. All of the connections to be made on a board are initially routed along the large open areas of the board, called spaces, and then a second pass of the algorithm finalizes the wiring within these spaces. None of the connections are finalized until all connections have been tentatively assigned and a great deal of flexibility exists for rerouting connections depending on localized routing conditions. The aperture used in channel routing, a space, is much larger than the aperture used by Lass. The basis of the channel routing algorithm is an efficient algorithm for assigning wire positions optimally within a large aperture which results in great speed. The storage requirements are kept low by using one word of storage (about 48 bits) for each line segment.

The channel routing algorithm is designed to be used in routing two-layer printed circuit boards which have the following properties. The types of component employed is a dual inline integrated circuit package. These packages are arranged on the printed circuit board in straight rows and columns (Fig. 1). Also there must be a capability for unlimited "floating" via holes. The positions of these vias are determined by the routing program since it assigns all vertical wire segments to one side of the board and all horizontal wire segments to the other. All connections between vertical and horizontal wire segments are made by means of via holes.

5.2.2 Spaces and Channels

The channel routing technique considers the printed circuit board to be subdivided into areas called spaces which are in turn subdivided into channels. A space is an area of a board which extends from one edge to the other and may be of any desired width. The channel routing technique uses spaces which cover the board as shown in Figure 12. The side of the board to which all vertical wire segments are assigned is divided into vertical spaces and the side to which all horizontal wire segments are assigned is divided into horizontal spaces. Each vertical space includes one column of package positions and half of the area on each side of that column. Each horizontal space includes one row of package pins along with half the area above and half the area below. This choice of space boundaries is significant because they are not restricted to fixed physical features of the board. For this reason space boundaries

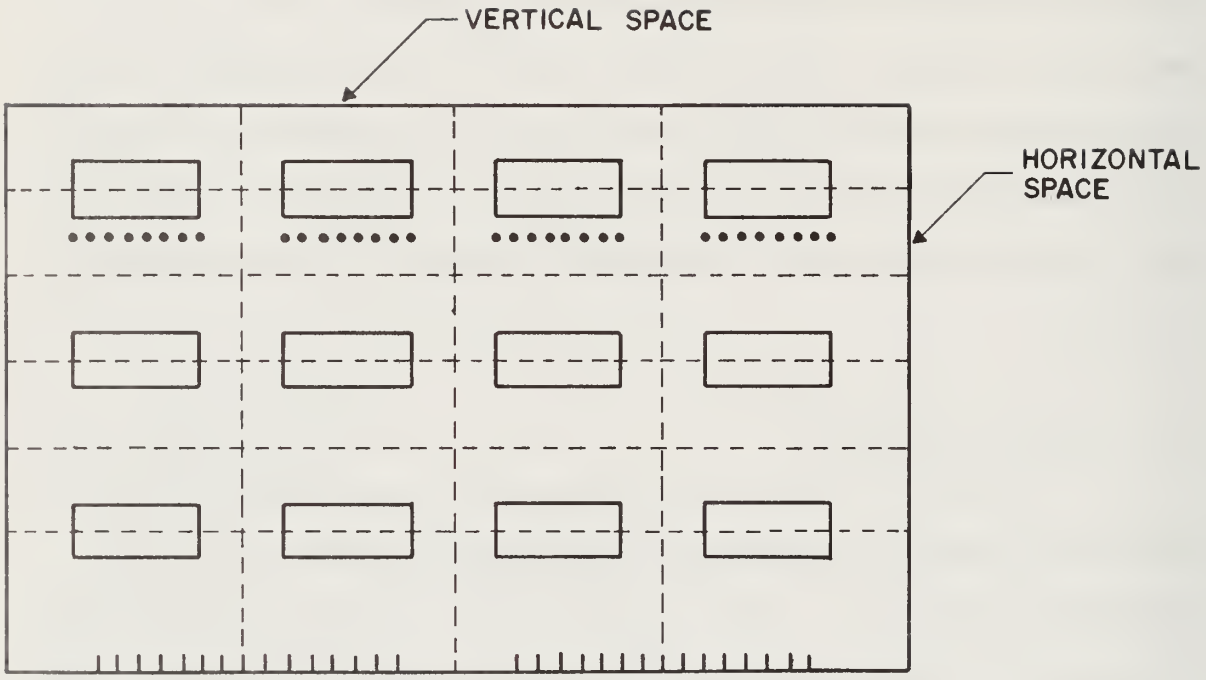


Figure 12. Space Arrangement.

can be repositioned quite easily depending on the amount of area needed to complete the wiring. Although there are obvious limitations, a maximal amount of flexibility is achieved while minimizing the complexity of the calculations that are required. (See 5.2.5.)

As a subdivision of a space, a channel also runs from one edge of the board to the other. The width of a channel is equal to the minimum spacing between parallel running wires. This means that a channel may contain more than one wire segment only if none of the wire segments overlap any other wire segment in that channel. Once all wire segments are assigned to spaces, an algorithm is applied to assign these wire segments to channels within the spaces.

5.2.3 Space Assignment

Each connection made on the board is specified by a starting pin and an object pin. The connection itself is in general broken up into five wire segments (Fig. 13). At each of the two pins a short vertical wire segment (A,E) is constructed which initially extends to the center of one of the adjoining horizontal spaces. The remaining vertical wire segment (C) can generally be assigned to one of several vertical spaces with the lengths of the horizontal wire segments (B,D) adjusted accordingly. In an attempt to evenly distribute vertical wire segments as much as possible, this wire segment (C) is always assigned to the vertical space between the starting and object pins which has the fewest wire segments already assigned to it. The specification of the two remaining horizontal wire segments (B,D) is thereby determined. Connections which are formed in the above manner will always be very close to minimal.

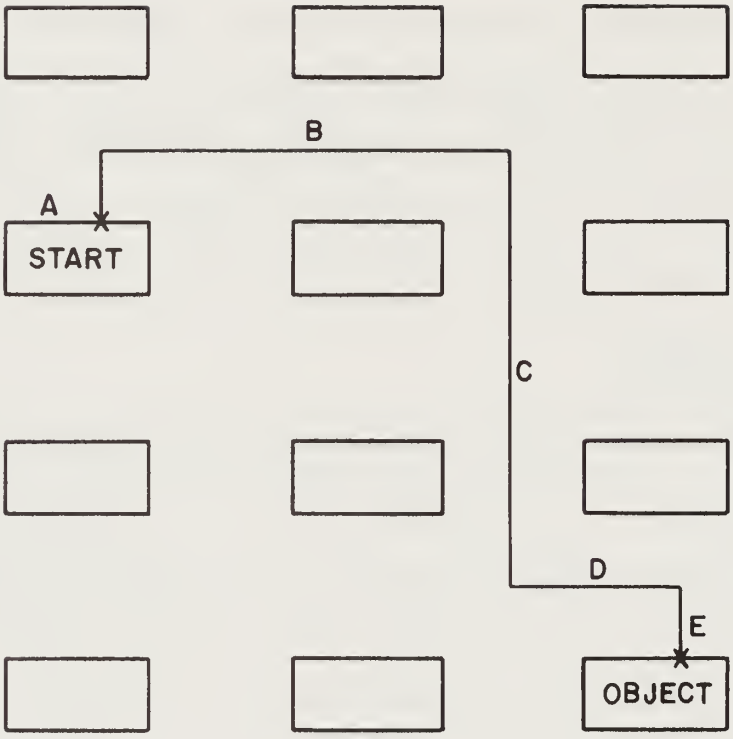


Figure 13. A Generalized Connection.

In fact, as long as all wire segments remain in the spaces they are originally assigned to, no connection will ever be more than 2V + 2H longer than its minimum. (Where V is the vertical distance between rows of packages and H is the horizontal distance between columns of packages.)

The channel routing program (Appendix B) stores the description of each wire segment in one word of memory. All of the horizontal wire segments are stored in one array and all the vertical wire segments in another. Each array row in one of these two-dimensional arrays corresponds to a space on the printed circuit board. Channel assignment within spaces is therefore concerned with only one array row at a time which allows for efficient partitioning to backup storage in a multiprogramming environment.

The description of a horizontal wire segment is detailed in Figure 14. Although 47 bits are indicated as being divided into eight fields, some of the fields are larger than necessary so that a reduced word size may be possible. The channel field is used to indicate which channel this wire segment is assigned to within its space. A channel field of six bits restricts the size of spaces to being less than 64 channels wide. The one bit indicator field, I, is initially set to zero and is changed to one when this wire segment is assigned to a channel.

Figure 14. Data Format of a Horizontal Wire Segment

The left pointer field contains a pointer to the vertical wire segment which connects (through a via) to the left end of this wire segment. Actually, the left pointer contains an index to a word of storage within the array row indicated by the left field. A left pointer field of nine bits allows a maximum of 512 wire segments to be assigned to each space. The right pointer field has the same purpose as the left pointer field, except that it refers to the right end point of the wire segment. The left field indicates the vertical space to which the left end point of this wire segment is restricted. A left field of five bits corresponds to a maximum of 32 vertical spaces being defined. The left offset field indicates the exact position of the left end point within the space indicated by the left field. The left offset field is changed after vertical channel assignment to indicate the channel to which the vertical wire segment that connects to the left end point is assigned. Therefore, the size of the left offset field is made consistent with the size of the channel field in reflecting the maximum number of channels per space. Combining the contents of the left field and the left offset field yields

one eleven-bit quantity which monetonically sequences the end points of all horizontal wire segments from right to left across the printed circuit board. Finally, the right field and the right offset field have the same purpose as the left and left offset fields respectively, except that they refer to the right end point of the wire segment.

Vertical wire segment information is also stored in words of memory which are divided into fields as inficated in Figure 14. The above explanation can be applied to the use of vertical wire segment fields by substituting the words, bottom and top, for the words, left and right, respectively and interchanging the words, vertical and horizontal.

When space assignment is complete all wire segments needed to completely wire the board will be represented by words of storage in the horizontal and vertical wire segment arrays. The final vertical wire segment on each end of a general connection, (segments A and E of Figure 13) do not require any storage however, because they do not need to be considered during channel assignments and can be easily specified later. Figure 15 shows that these short final vertical wire segments do not take up any vertical channel space, and since no two pins have the same horizontal coordinate within a horizontal space, these final wire segments can not conflict with one another. Specifying that the end point of a horizontal wire segment is to be connected to a pin, (done by setting the pointer field to all 1's) and knowing the horizontal coordinate of that end point is all that is required to completely describe the vertical wire segment needed to complete that connection.

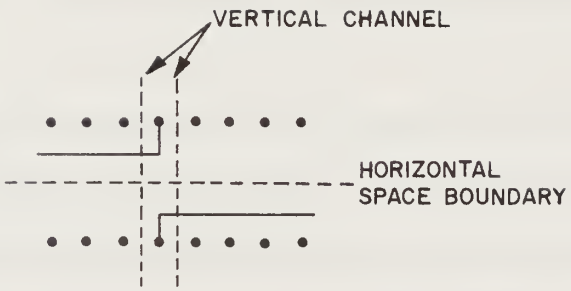


Figure 15. Wiring to a Package Pin.

5.2.4 Channel Assignment

After space assignment is completed each space will contain a set of wire segments. These wire segments are represented as a set of intervals having an upper bound and a lower bound (Fig. 16). The object of channel assignment is to position all wire segments so that no two overlap and to use the fewest possible number of channels. The first step in the procedure is to search the list of intervals for the element which has the greatest upper bound. This element is assigned to the first channel and eliminated from the list. The list is then searched for the interval which has the greatest upper bound which is less than the lower bound of the previously chosen interval. This element is also assigned to the first channel and eliminated from the list. The search is repeated until none of the remaining elements have an upper bound which is less than the previous lower bound, at which time the entire process is repeated for the next channel. When all of the intervals have been eliminated from the list the channel assignment is complete. This algorithm always uses the minimum possible number of channels to finalize the positions of all of the wire segments in a given space. The proof of minimality will be given in Section 5.2.6.

When the channel assignments are being made, precautions must be taken to ensure that the wire segments which must be joined to form a connection have end points which coincide. At the conclusion of the space assignment stage, many conflicts may exist since all wire segments are assumed to travel down the center of the space. Assume that two horizontal wire segments (1 and 2) have the same end point (Fig. 17a). If

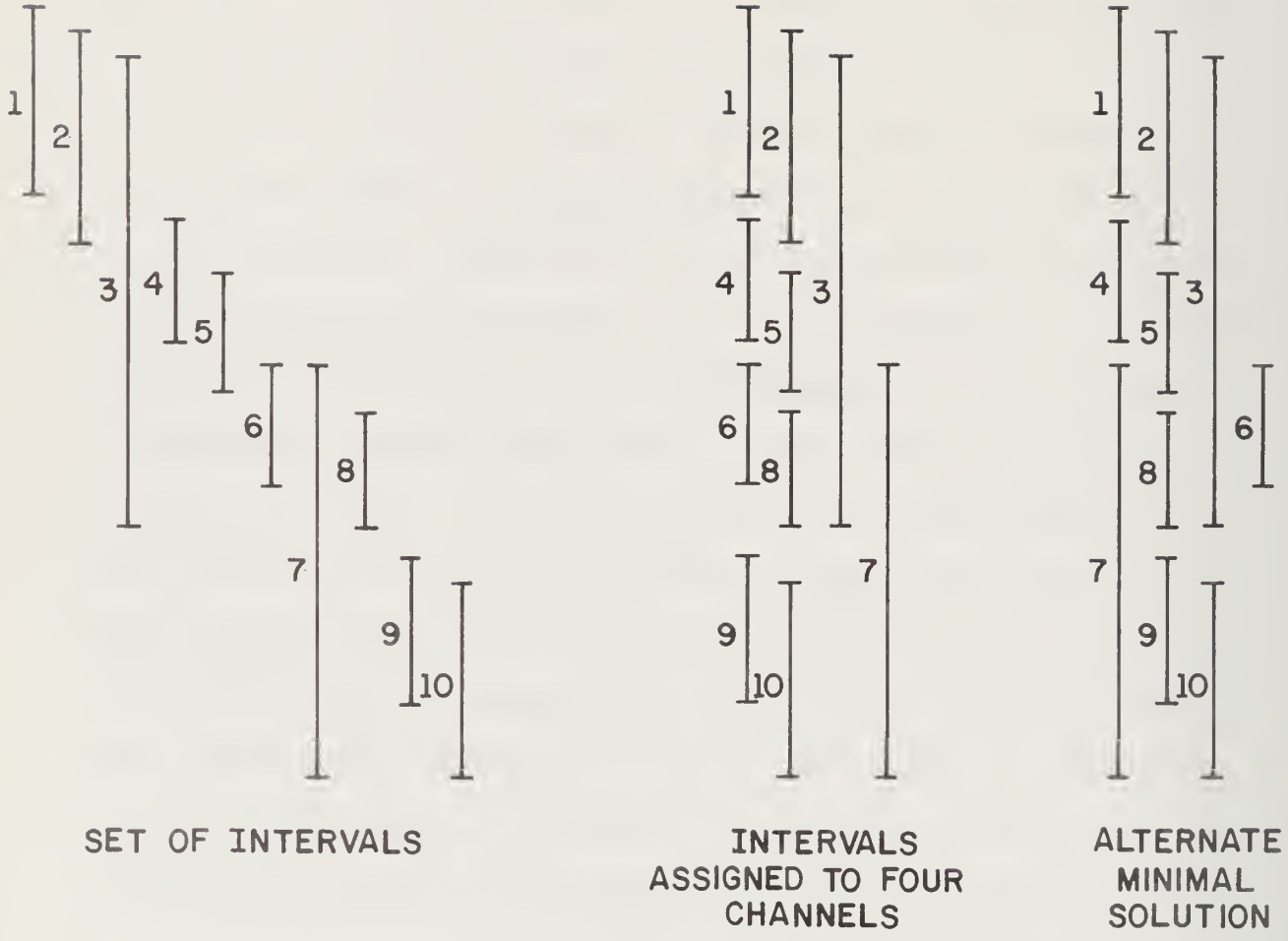


Figure 16. Channel Assignment.

vertical wire segment 3 is to connect to 2 and vertical wire segment 4 is to connect to 1 then the two connections are sharing one via hole. Channel assignment is now performed on all vertical wire segments. Since the upper bound of wire 4 is not less than the lower bound of wire 3, these two wire segments may not be assigned to the same channel. For this reason there can no longer be a via conflict between connections 14 and 23, but wire 1 may be in conflict with wire 2 (Fig. 17b). If such a conflict exists then wire 1 and wire 2 can not be placed in the same channel when horizontal channel assignment is performed. When all channel assignment is complete there is no conflict between connection 14 and connection 23 (Fig. 17c). Other combinations of conflicting wire segments can be formed, but the channel assignment proceeds in such a way as to always eliminate wire overlap and via conflicts.

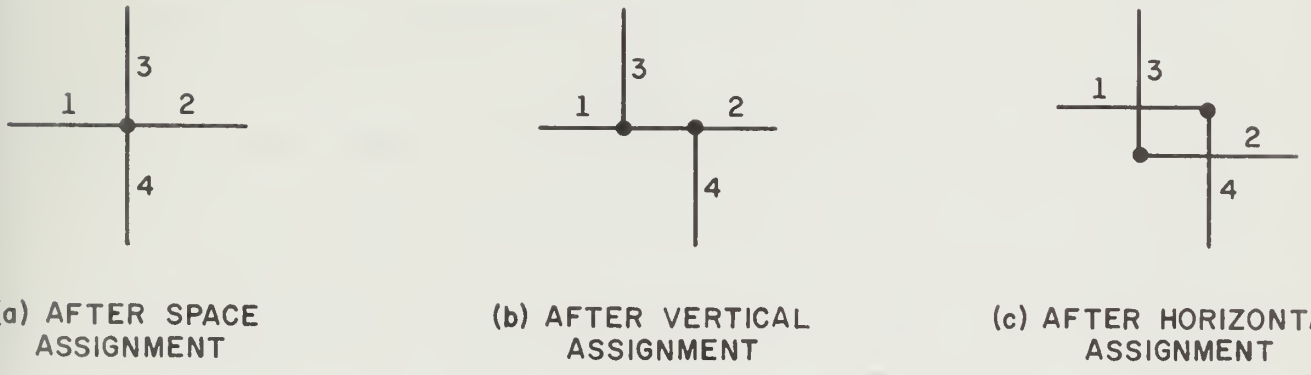


Figure 17. Conflict Elimination.

5.2.5 Final Positioning

After channel assignment has been completed in each space, the number of channels needed for complete wiring is determined for each space. In some cases the number of channels needed may surpass the number of channels originally allocated to a given space. Although the availability of wiring area is restricted by the physical layout of packages on the board, the space boundaries are not tied to physical positions as explained in 5.2.2. Figure 18a shows a typical initial positioning of space boundaries. The channel assignment (Fig. 18b) appears to overflow space B which would cause incomplete wiring. However, by moving the boundary between space A and space B upward one channel it becomes possible to fit all wires without overflow (Fig. 18c).

Once channel assignment has been completed a simple algorithm can be applied to optimally position the horizontal space boundaries. The objective of this procedure is to minimize the number of channels which will not fit within the physical limitations over the entire horizontal side of the board. This algorithm consists merely of locating each boundary between spaces to its upward most possible position (i.e., farthest from the edge connectors on the board). A space boundary can be moved upward until it encounters either an already fixed space boundary or a physical limitation (either a row of pins or the edge of the board). Starting from the uppermost boundary of the space the channels assigned to that space are given their final positions working downward until either the set of channels for that space is exhausted, or the lower physical boundary for that space is encountered, at which time the remaining

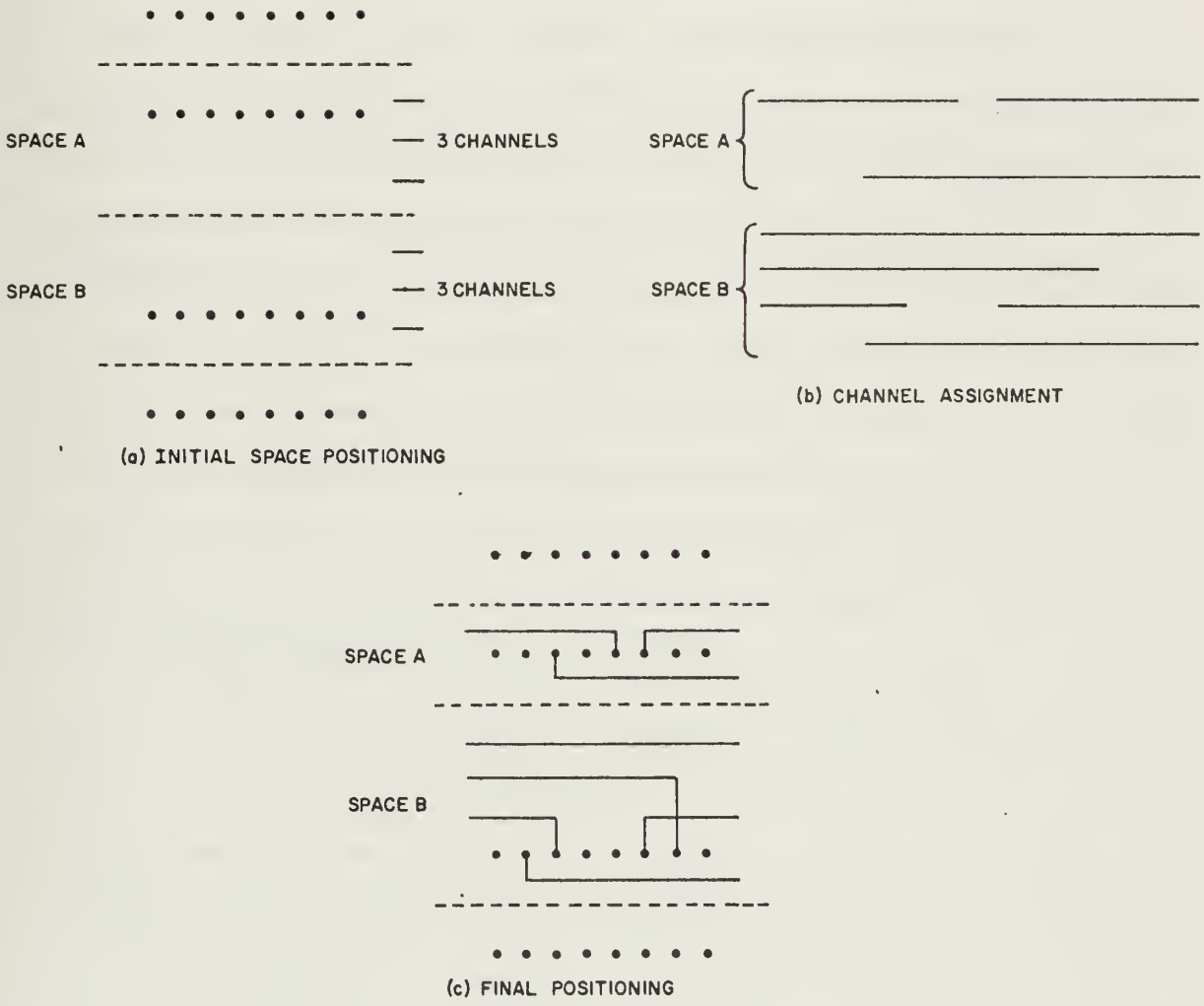


Figure 18. Final Horizontal Positioning.

channels are entered in an overflow list. In either case a downward boundary is set for the space which may serve as the upward boundary for the next space. This procedure will minimize the number of channels which will be entered in the overflow list.

The final positioning of channels requires an update to the positions of the horizontal wire segments within each channel. Also, an update must be made to the endpoints of vertical wire segments which connect to the horizontal wire segments that are moved. These updates are quite easily made. Also, it can be easily verified that these adjustments due to final positioning will not create conflicts and will not effect the optimality of the original channel assignments that were made.

The same algorithm can be applied to the final positioning of vertical channels since vertical space boundaries are also free to move within physical limitations. However, in the case of final vertical positioning, changes may be caused in horizontal wire segments which will effect horizontal channel assignment. In Figure 19 the movement of wire segment A causes wire segment B to change orientation and therefore conflict with wire segment C. Such a change in wire segment orientation can cause a change in the channel assignments. In Figure 19a, wire segments B and C do not conflict, and could be assigned to the same channel, whereas, in Figure 19b, wire segments B and C can not be assigned to the same channel. A straight forward solution to this problem has been adopted. Vertical channel assignment and final positioning are performed before horizontal channel assignment begins. Final horizontal positioning and adjustments then completes the wire routing process.

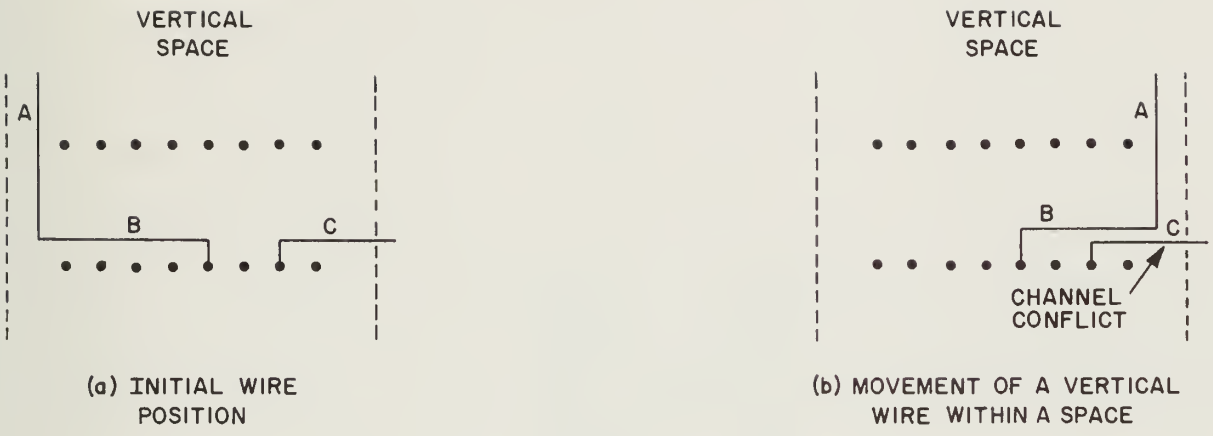


Figure 19. Final Vertical Positioning.

5.2.6 Proof of Optimum Channel Assignment

The set of wire segments assigned to a given space can be represented as a set of intervals as previously stated (Fig. 16). This set of intervals can be represented by a directed graph (Fig. 20). The nodes of the graph represent wire segments and an arrow is directed from one node to another if and only if the lower bound of the first wire is greater than the upper bound of the other wire. Assigning wire segments to channels corresponds to covering the directed graph which represents the set of wire segments with a set of directed paths. A directed path is composed of a set of nodes which are connected by arrows such that all arrows point in the direction of the path (dark arrows in Figure 20). Selecting a path corresponds to assigning the wire segments represented by the nodes

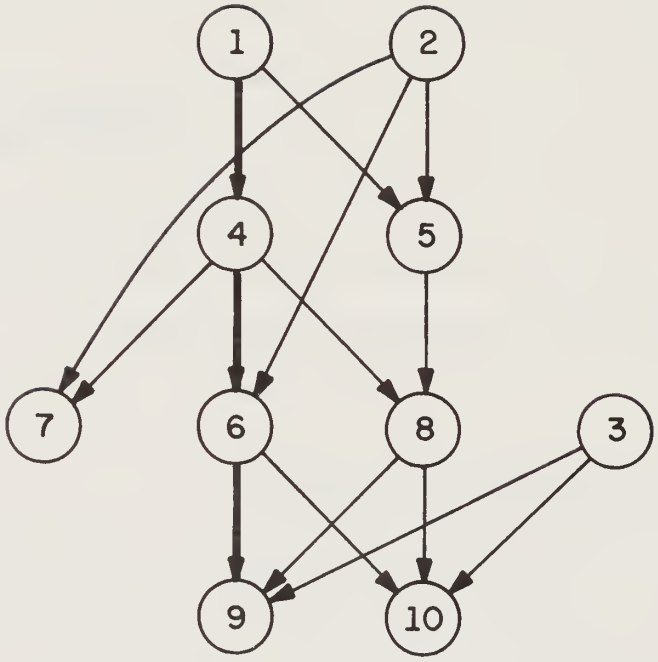


Figure 20. A Directed Graph.

on the path to one channel without overlap. A path cover of a graph is a collection of paths which includes each node of the graph exactly once. Two nodes in a directed graph are said to be incomparable if no path exists which includes both nodes. The greatest number of mutually incomparable nodes which can be found in the directed graph must therefore be equal to the minimum number of paths which can be used to cover the graph. This property was first proven by Dilworth and the corresponding theorem bears his name. The largest collection of mutually incomparable nodes is referred to as a maximal incomparable set and in general several distinct incomparable sets from one graph may be maximal.

To show that the channel assignment algorithm is optimal it must be shown that the number of channels used is equal to the minimum number of paths which can cover the corresponding directed graph. First of all a method of constructing paths must be found which corresponds to the method used to assign wire segments to channels. Such a path can be formed by choosing the node with the greatest upper bound then choosing the arrow eminating from that node which leads to the node with the greatest upper bound. The path is complete when a node is reached which has no outgoing arrows. The first step in the proof is to show that such a path which will be called a max chain contains exactly one member from each maximal incomparable set of the graph. By definition no path can contain more than one element of a maximal comparable set of nodes. Assume on the other hand that a max chain can be found which contains no members of a given maximal incomparable set. If this assumption leads to a contradiction then the first step is demonstrated. Since none of

the nodes (N_1, ..., N_p) are in a given maximal incomparable set (M_1, ..., M_q) they must all be comparable with a member of that set (Fig. 21). There is a node N_j which is the highest node with an incoming arrow which leads from a member M_k of M. The next preceding node N_i must have an outgoing arrow which leads to a member M_e of M. From the comparisons described above it follows that the lower bound of M_k is greater than the upper bound of N_j and the upper bound of M_e is greater than the lower bound of M_k. Therefore the upper bound of M_e is greater than the upper bound of N_j. This proves that N is not a max chain since the arrow from N_i to M_e was not chosen. Therefore every max chain contains exactly one member from each maximal incomparable set. Eliminating the nodes along a max chain from the directed graph reduces the size of each maximal incomparable set by one, reducing the number of paths required to cover the graph by one. Therefore the number of max chains needed to cover the graph which is equal to the number of channels used is always the minimum.

This algorithm assumes that each wire segment is an indivisible entity. The question arises as to whether fewer channels would be required if a wire segment could be split into independent subsections connected by perpendicular wire segments. To show that such a modification would not improve the number of channels required a proof will be given that the size of the maximal incomparable sets is not decreased. Assume that a wire segment which is a member of a given maximal incomparable set can be broken into several nodes so that none are incomparable with the maximal incomparable nodes. The lower bound of any of these nodes would

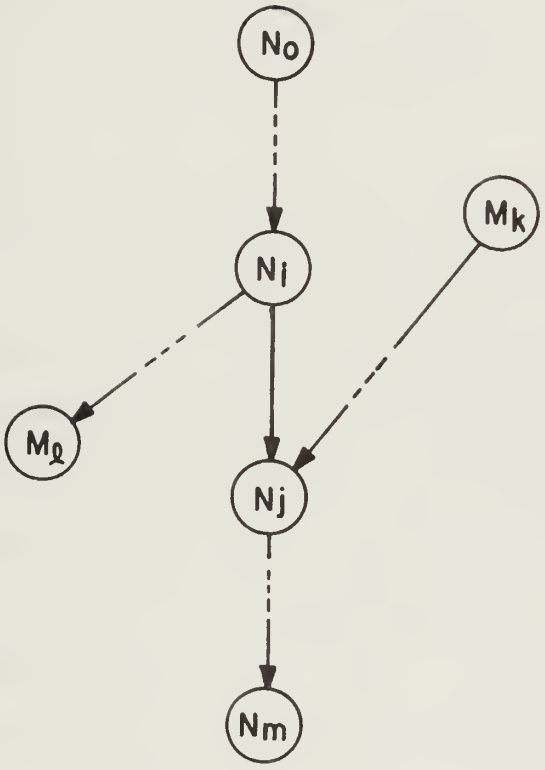


Figure 21. A Generalized Path.

be equal to the upper bound of the following node to preserve continuity. Figure 11 can again be used to show that a contradiction is generated. Node M_k is shown comparable with node M_e ($L(M_k) > U(N_j) > U(M_e)$). Therefore such a division of one wire segment can not reduce the number of channels required to place a set of wire segments.

5.2.7 Results

The Channel Routing program (Appendix B) was run on 20 ILLIAC IV CU boards placed by the Serial Placement program 4.2.3. The results of routing within the package spacing shown in Figure 4 are given in Table 7. The density of wiring is presented as the percent of board area used. This figure can be calculated accurately only for those boards which are completely wired. The amount of board space available is considered to be the total area up to the farthest package row from the connector pins which has any packages in it. This is a reasonable choice, since the Channel Routing program never routes any wires outside this area. The wiring density for boards which are not completely wired has a different meaning. This figure gives the density that would be attained if the package spacings could be increased to allow complete wiring.

The speed of the Channel Routing program is its most important assests. Many improvements of the type mentioned in Section 5.2.8 can be made without raising the running time above even one minute. The channel assignment routine which is the basis of the program takes only two or three seconds to finalize the wiring. Most of the program time is spent setting up the wire segment storage. This includes reading the wire list

Table 7. Channel Routing Results

Board Name	Total Wire Length	Vertical Overflow	Horizontal Overflow	Routing Time (sec)	Percent of Board Area Used
FIRDB	725	0	1	8	19*
FDS	1102	0	0	11	30
IWSCTL	1159	0	0	11	28
FDQA	1230	0	0	10	23
FIRCB	1292	0	0	12	25
FORC	1313	0	0	12	25
IIAA	1340	0	0	12	21
FWLDF	1346	0	0	10	26
ATP07	1429	0	4	13	29*
MTMCTL	1460	0	0	11	25
MPCTL	1527	0	0	14	29
TXFER	1608	0	5	12	31*
ICTLA	1637	0	0	12	26
MDSPLY	1736	16	32	13	36*
ATP05	1851	43	0	17	21*
FBUSY	2058	24	3	15	32*
ATP25	2126	0	0	17	34
TCRFLD	2318	16	9	15	33*
ILTCL	2416	37	0	17	34*
IICR1	3306	42	78	20	35*

* These percentages are calculated by increasing the board dimensions to those necessary for 100 percent wiring.

and performing the space assignment. Also, the data structure of the program is well suited to a small core, multiprogramming computer. The channel assignment routine is basically interested in one array row at a time which keeps data storage requirements in core below 1000 words. The entire program requires about 5000 words of resident core to run on a Burroughs 6700 computer.

5.2.8 Possible Improvements

Two alternatives exist for improving vertical space assignment. One is to run the entire routing program once, simply to determine which vertical spaces use too many channels. This information can then be used to allocate space better on a second routing attempt. Another possibility is to perform vertical channel assignment is still in progress. This can be easily done because vertical wire segments can initially end only at the center of horizontal spaces. A simple count kept at each end point position will reflect the number of channels required so far. A combination of both of these improvements might be better still.

Changes in the final positioning routine may produce the most dramatic improvements. Final positioning allows space boundaries to be moved within certain physical limits. An improvement would be to allow wire segments to actually cross the space boundary as long as no conflicts are created. Rather than having wire segments cross space boundaries, the identical result can be attained by allowing space boundaries which are not straight lines. Preliminary study of routing data indicates that such a modification may increase wiring densities by five percent. More sophisticated final positioning algorithms can be developed and should be investigated.

In the event of wiring failures a follow-up technique should be considered which takes advantage of the special properties of the channel routing method. The storage organization and the high speed of the channel routing program should make failure elimination possible without extensive use of maze running algorithms. One possibility is to change a number of space assignments so as to reduce localized overcrowding and then return to the original channel assignment routine.

6. CONCLUSIONS

6.1 Placement

Several basic placement algorithms were compared in Chapter 4. During the course of implementing these algorithms some factors were found to greatly influence the results. The pairwise interchange method and Steinberg's algorithm were initially programmed with approximated wire length measurements. No significant improvements were made in the initial serial placements. After these two programs were changed so that all wire-lengths were exactly measured, they were able to make considerable improvements on the initial placements. Also, it was found, as suggested by Garside and Nicholson [29], that systematic pairwise interchange converges faster than random pairwise interchange. These conclusions are not intended as comparisons of alternative implementations. The factors involved are important because it was found that the algorithms simply would not work unless they were handled as indicated.

The results presented in Chapter 4 support several interesting conclusions. The total wire length of a board must be augmented by other measures to clearly indicate the ease of wiring that board. Connections which leave the board (edge connector pins) significantly influence the wireability of the board. The most important measure however, is the distribution of wiring density. These results indicate that assignment of edge connector pins is very important and better results can be obtained by allowing the placement program to assign the pin positions. Also, an effort should be made to incorporate information from a routing attempt into a placement improvement program. Such an effort is becoming practical with the introduction of fast wire routing techniques like the channel routing algorithm of Chapter 5.

The comparison of serial and iterative placement techniques presented in Table 6 yields several conclusions. The serial placement method has a distinct time advantage and may produce completely wireable placements. In such cases, no further improvement should be sought. Iterative placement algorithms can produce improvements in wire length and wireability when needed without using an unreasonable amount of time. The iterative placement results of Chapter 4 were obtained with the possible placement positions restricted to those used for serial placement. If an iterative placement algorithm were allowed to reposition components to any position on the board, even better results should be attainable. Further comparisons using sophisticated versions of existing placement algorithms will provide important qualitative and quantitative conclusions.

6.2 Channel Routing

The channel routing algorithm described in Chapter 5 performs the complete wire routing function. All connections which can be made by this algorithm within the given board area are completely specified without conflicts. The results of implementing this basic algorithm and testing it on ILLIAC IV design problems are very favorable. The speed of execution for such large problems is the main advantage of the channel routing approach. The major disadvantage is that this algorithm is dependent on the availability of unlimited via holes whose positions are not fixed.

An implementation of the basic channel routing algorithm is capable of completely wiring large printed circuit boards with wiring densities of 20 to 30 percent. Wiring failures begin to occur on boards

requiring densities greater than 30 percent. These results are significant because several improvements can be made in the basic algorithm. More sophisticated space assignment and final positioning routines can be implemented. Also a second pass algorithm could be developed to route as many failed connections as possible using more complicated paths than the channel routing algorithm presently considers. The speed and flexibility of the basic algorithm will make improvements possible without unreasonable costs in terms of computer time.

APPENDIX A

The following pages contain the signal net listings of five ILLIAC IV CU boards. Several simplifications of the original net lists have been made, including the elimination of all power and ground connections and a consolidation of the clock buffer circuitry so that it appears as a single movable package. Also, the labeling of edge connector pins has been organized to facilitate distance measurements for placement programs and edge pin wiring for routing programs. The edge connectors should be considered to be divided as shown in Figure 1 with the groups being labeled P001 through P015. Within each group there are exactly sixteen pins which are spaced 50 mils apart and are numbered as shown in Figure 5, so each group can be treated the same as a package. Actually each edge connector group has 32 pins with sixteen appearing on each side. In the signal netlists which follow, no distinction will be made between corresponding edge connector pins on opposite sides of the board. Therefore, one edge connector pin name may appear more than once in the netlist. However, no edge connector pin is used more than once, so each occurrence should be treated as a separate pin.

The netlists are listed in columns so that the first entry of the second column follows the final entry of the first column and so forth. Each entry is composed of 1) a name where A indicates a package and P indicates a connector pin group, 2) a pin number and 3) a source (S) or load (L) indicator. Each signal net is composed of one source and one or more loads which immediately follow the source. The following rules should be helpful in checking that the data is properly copied.

1. Packages are named starting with A000 sequentially through the highest numbered package. No sequence numbers are skipped.
2. A package name with sequence number N will not appear in a given entry unless all package names with sequence numbers less than N have appeared in previous entries.
3. Edge connector pin groups are named P001 through P015 inclusive, and no sequence number greater than 15 will follow a "P".
4. Pin numbers (the second element of each entry) must be between 001 and 016 inclusive.
5. Each netlist must begin with a source (an entry with an S as its third element) and every source (S) must be followed by at least one load (L).
6. Each column after the first page has 50 entries (49 on first page).

Netlist	Highest Numbered Package	Length of Netlist
ATP07	A100	960
FDQA	A092	861
FIRDB	A051	484
IICR1	A120	1233
TCRFLD	A135	995

ATP07

A000 005 S
A001 008 L
A002 005 S
A003 008 L
A000 002 S
A001 011 L
A002 002 S
A003 011 L
A004 002 S
A005 001 L
A006 002 S
A007 001 L
A008 013 L
A009 002 S
A003 014 L
A010 013 L
A011 008 L
A012 002 S
A013 014 L
A002 014 L
A010 011 L
A011 013 L
A014 002 S
A015 001 L
A016 002 S
A017 001 L
A018 013 L
A019 002 S
A001 014 L
A020 013 L
A021 008 L
A022 002 S
A023 014 L
A000 014 L
A020 011 L
A021 013 L
A024 002 S
A017 008 L
A025 002 S
A007 008 L
A026 007 S
A027 008 L
A026 004 S
A027 009 L
A028 007 S
A027 011 L
A028 004 S
A027 012 L
A029 007 S
A027 013 L
A029 004 S
A027 014 L
A030 007 S
A027 016 L
A030 004 S
A027 001 L
A031 007 S
A032 008 L
A031 004 S
A032 009 L
A033 007 S
A032 011 L
A033 004 S
A032 012 L
A034 007 S
A032 013 L
A034 004 S
A032 014 L
A035 007 S
A032 016 L
A035 004 S
A032 001 L
A036 007 S
A037 008 L
A036 004 S
A037 009 L
A038 007 S
A037 011 L
A038 004 S
A037 012 L
A039 007 S
A037 013 L
A039 004 S
A037 014 L
A040 007 S
A037 016 L
A040 004 S
A037 001 L
A041 007 S
A042 008 L
A041 004 S
A042 009 L
A043 007 S
A042 011 L
A043 004 S
A042 012 L
A044 007 S
A042 013 L
A044 004 S
A042 014 L
A045 007 S
A042 016 L
A045 004 S
A042 001 L
A020 005 S
A017 011 L
A010 005 S
A007 011 L
A004 007 S
A008 012 L
A008 014 L
A011 001 L
A011 011 L
A046 013 L
A006 007 S
A025 016 L
A010 009 L
A010 014 L
A008 011 L
A011 009 L
A011 016 L
A046 011 L
A009 007 S
A002 011 L
A002 016 L
A025 014 L
A010 008 L
A008 009 L
A011 014 L
A047 013 L
A012 007 S
A013 013 L
A002 009 L
A025 013 L
A008 008 L
A047 011 L
A014 007 S
A018 012 L
A018 014 L
A021 001 L
A021 011 L
A048 013 L
A016 007 S
A024 016 L
A020 009 L
A020 014 L
A018 011 L
A021 009 L
A021 016 L
A048 011 L
A019 007 S
A000 011 L
A000 016 L
A024 014 L
A020 008 L
A018 009 L
A021 014 L
A049 013 L
A022 007 S
A023 013 L
A000 009 L
A024 013 L
A018 008 L
A049 011 L
A050 007 S
A051 011 L
A052 011 L
A050 005 S
A051 013 L
A052 013 L
A050 004 S
A053 011 L
A054 011 L
A050 002 S
A053 013 L
A054 013 L
P003 015 S
A055 011 L
A055 007 S
P004 002 L
A055 008 S
A050 001 L
A050 009 L
A020 002 S
A017 013 L
A010 002 S
A007 013 L
A018 002 S
A015 008 L
A008 002 S
A005 008 L
A021 005 S
A015 011 L
A011 005 S
A005 011 L
A021 002 S

A015 013 L
A011 002 S
A005 013 L
P004 015 S
A056 009 L
A056 014 L
A056 004 S
A057 011 L
A058 011 L
A056 002 S
A057 013 L
A058 013 L
A056 007 S
A059 011 L
A060 011 L
A056 005 S
A059 013 L
A060 013 L
A061 001 S
A056 008 L
A061 002 S
A050 008 L
A061 003 S
A056 013 L
A061 004 S
A050 013 L
A007 005 S
A045 011 L
A045 016 L
A007 007 S
A044 011 L
A044 016 L
A003 005 S
A043 011 L
A043 016 L
A003 007 S
A041 011 L
A041 016 L
A013 002 S
A035 011 L
A035 016 L
A013 004 S
A034 011 L
A034 016 L
A017 005 S
A040 011 L
A040 016 L
A017 007 S
A039 011 L
A039 016 L

A001 005 S
A038 011 L
A038 016 L
A001 007 S
A036 011 L
A036 016 L
A023 002 S
A030 011 L
A030 016 L
A023 004 S
A029 011 L
A029 016 L
P009 016 S
A048 009 L
A048 014 L
A047 009 L
A047 014 L
A046 009 L
A046 014 L
A049 009 L
A049 014 L
P011 006 S
A048 012 L
A047 012 L
A046 012 L
A049 012 L
P013 003 S
A028 011 L
A028 016 L
P009 012 S
A026 011 L
A026 016 L
A023 012 L
A001 009 L
A017 009 L
P013 004 S
A033 011 L
A033 012 L
P009 007 S
A031 011 L
A031 016 L
A013 012 L
A003 009 L
A007 009 L
A015 005 S
P008 011 L
A015 007 S
P008 016 L
A005 007 S
P009 015 L

P007 003 S
A062 012 L
A063 012 L
P007 004 S
A064 001 L
P005 011 S
A065 002 L
A042 002 S
A046 016 L
A046 004 S
P010 011 L
A042 005 S
A046 008 L
A046 005 S
P010 016 L
A032 002 S
A047 016 L
A047 004 S
P011 003 L
A032 005 S
A047 008 L
A047 005 S
P010 015 L
A037 002 S
A048 016 L
A048 004 S
P010 014 L
A037 005 S
A048 008 L
A048 005 S
P011 004 L
A027 002 S
A049 016 L
A049 004 S
P011 007 L
A027 005 S
A049 008 L
A049 005 S
P011 003 L
A018 007 S
P009 016 L
A018 005 S
P009 011 L
A008 005 S
P008 012 L
A066 005 S
A067 011 L
A066 004 S
A067 013 L
A068 005 S

A069 011 L
A068 004 S
A069 013 L
A070 005 S
A071 011 L
A070 004 S
A071 013 L
A072 005 S
A073 011 L
A072 004 S
A073 013 L
P004 015 S
A067 012 L
A067 014 L
A069 012 L
A069 014 L
A071 012 L
A071 014 L
A073 012 L
A073 014 L
A074 013 S
P003 003 L
A051 007 S
P004 004 L
A051 008 S
A074 011 L
A075 013 S
A003 004 L
A053 007 S
A066 009 L
P004 007 L
A053 008 S
A075 011 L
A075 016 S
P002 015 L
A053 002 S
A066 016 L
P003 015 L
A053 001 S
A075 014 L
A076 013 S
P003 007 L
A054 007 S
A068 009 L
P003 016 L
A054 008 S
A076 011 L
A076 016 S
P002 011 L
A054 002 S

A068 016 L
P003 012 L
A054 001 S
A076 014 L
A074 016 S
P002 012 L
A051 002 S
A070 016 L
A051 001 S
A074 014 L
A077 013 S
P003 003 L
A052 007 S
A072 009 L
A052 008 S
A077 011 L
A077 016 S
P002 016 L
A052 002 S
A072 016 L
A052 001 S
A077 014 L
P005 015 S
A070 011 L
A072 011 L
P006 002 S
A066 011 L
A068 011 L
A062 007 S
A051 012 L
A078 005 S
A053 012 L
A079 005 S
A053 014 L
A064 005 S
A054 012 L
A065 005 S
A054 014 L
A062 002 S
A051 014 L
A063 007 S
A052 012 L
A063 002 S
A052 014 L
A080 007 S
A066 008 L
A080 005 S
A081 014 L
P011 008 L
A080 004 S

A066 001 L
A080 002 S
A081 012 L
P009 015 L
A082 007 S
A068 008 L
A082 005 S
A083 014 L
P011 002 L
A082 004 S
A068 001 L
A082 002 S
A083 012 L
P007 004 L
A084 007 S
A070 008 L
A084 005 S
A085 014 L
P007 007 L
A084 004 S
A070 001 L
A084 002 S
A085 012 L
P007 008 L
A086 007 S
A072 008 L
A086 005 S
A087 014 L
P007 011 L
A086 004 S
A072 001 L
A086 002 S
A087 012 L
P007 008 L
P004 016 S
A066 007 L
A068 007 L
A070 007 L
A072 007 L
P008 004 S
A087 011 L
A087 013 L
A085 011 L
A085 013 L
A083 011 L
A083 013 L
A081 011 L
A081 013 L
P014 007 S
A088 011 L

P014 003 S
A089 011 L
P014 004 S
A090 011 L
P014 008 S
A091 011 L
P013 015 S
A092 011 L
P014 007 S
A093 011 L
P014 003 S
A094 011 L
P013 016 S
A095 011 L
P010 007 S
A080 009 L
P013 011 S
A088 013 L
P010 003 S
A080 014 L
P013 015 S
A089 013 L
P010 004 S
A082 009 L
P013 012 S
A090 013 L
P010 003 S
A082 014 L
P013 016 S
A091 013 L
P010 008 S
A084 009 L
P014 004 S
A092 013 L
P010 011 S
A084 014 L
P013 007 S
A093 013 L
P010 007 S
A086 009 L
P013 011 S
A094 013 L
P010 012 S
A086 014 L
P013 008 S
A095 013 L
P014 011 S
A088 012 L
A089 012 L
A090 012 L

A091 012 L
A092 012 L
A093 012 L
A094 012 L
A095 012 L
P014 016 S
A088 014 L
A089 014 L
A090 014 L
A091 014 L
A092 014 L
A093 014 L
A094 014 L
A095 014 L
P009 004 S
A078 009 L
A064 011 L
A079 012 L
A065 011 L
P008 003 S
A062 009 L
A062 014 L
A063 009 L
A063 014 L
P008 008 S
A078 014 L
A064 016 L
A079 014 L
A065 014 L
P007 012 S
A062 008 L
P008 007 S
A080 008 L
A078 013 L
P005 012 S
A080 013 L
A079 013 L
P007 016 S
A082 008 L
A064 014 L
P005 003 S
A082 013 L
A065 016 L
P010 015 S
A084 008 L
P007 015 S
A084 013 L
A062 016 L
P007 015 S
A086 008 L

A063 008 L
P005 007 S
A086 013 L
A063 016 L
P009 003 S
A078 008 L
P006 011 S
A079 011 L
P007 011 S
A064 009 L
P006 007 S
A065 009 L
P007 003 S
A062 011 L
P008 003 S
A088 008 L
A078 016 L
P005 011 S
A089 008 L
A079 016 L
P007 012 S
A090 008 L
A064 012 L
P005 004 S
A091 008 L
A065 012 L
P014 011 S
A092 008 L
P008 004 S
A093 008 L
A062 013 L
P007 016 S
A094 008 L
A063 011 L
P005 003 S
A095 008 L
A063 013 L
P008 015 S
A078 011 L
P006 015 S
A079 008 L
P007 007 S
A064 007 L
P006 003 S
A065 007 L
P008 011 S
A078 012 L
A064 008 L
A079 009 L
A065 008 L

P008 007 S
A078 001 L
A064 013 L
A079 001 L
A065 013 L
P014 012 S
A088 009 L
A089 009 L
A090 009 L
A091 009 L
A092 009 L
A093 009 L
A094 009 L
A095 009 L
P012 015 S
A044 013 L
A045 008 L
A004 012 L
P011 011 S
A044 008 L
A045 013 L
A004 009 L
P012 011 S
A041 013 L
A043 008 L
A006 012 L
P011 012 S
A041 008 L
A043 013 L
A006 009 L
P012 012 S
A034 013 L
A035 008 L
A009 012 L
P011 015 S
A034 008 L
A035 013 L
A009 009 L
P011 007 S
A031 013 L
A033 008 L
A012 012 L
P011 011 S
A031 008 L
A033 013 L
A012 009 L
P012 016 S
A039 013 L
A040 008 L
A014 012 L

P011 016 S
A039 008 L
A040 013 L
A014 009 L
P012 007 S
A036 013 L
A038 008 L
A016 012 L
P012 003 S
A036 008 L
A038 013 L
A016 009 L
P012 015 S
A029 013 L
A030 008 L
A019 012 L
P011 015 S
A029 008 L
A030 013 L
A019 009 L
P012 011 S
A026 013 L
A028 008 L
A022 012 L
P011 004 S
A026 008 L
A028 013 L
A022 009 L
A088 005 S
A044 014 L
A045 014 L
A004 011 L
A004 014 L
A088 007 S
A044 009 L
A045 009 L
A004 008 L
A089 005 S
A041 014 L
A043 014 L
A006 011 L
A006 014 L
A089 007 S
A041 009 L
A043 009 L
A006 008 L
A090 005 S
A034 014 L
A035 014 L
A009 011 L

A009 014 L
A090 007 S
A034 009 L
A035 009 L
A009 008 L
A091 005 S
A031 014 L
A033 014 L
A012 011 L
A012 014 L
A091 007 S
A031 009 L
A033 009 L
A012 008 L
A092 005 S
A039 014 L
A040 014 L
A014 011 L
A014 014 L
A092 007 S
A039 009 L
A040 009 L
A014 008 L
A093 005 S
A036 014 L
A038 014 L
A016 011 L
A016 014 L
A093 007 S
A036 009 L
A038 009 L
A016 008 L
A094 005 S
A029 014 L
A030 014 L
A019 011 L
A019 014 L
A094 007 S
A029 009 L
A030 009 L
A019 008 L
A095 005 S
A026 014 L
A028 014 L
A022 011 L
A022 014 L
A095 007 S
A026 009 L
A028 009 L
A022 008 L

A067 005 S
P006 016 L
A067 004 S
P006 015 L
A069 005 S
P006 012 L
A069 004 S
P006 011 L
A071 005 S
P006 008 L
A071 004 S
P006 007 L
A073 005 S
P006 004 L
A073 004 S
P006 006 L
A081 004 S
A059 012 L
A081 005 S
A059 014 L
A083 004 S
A060 012 L
A083 005 S
A060 014 L
A085 004 S
A057 012 L
A085 005 S
A057 014 L
A087 004 S
A058 012 L
A087 005 S
A058 014 L
A096 013 S
P005 015 L
A059 007 S
A066 012 L
A059 008 S
A096 011 L
A096 016 S
P006 003 L
A059 002 S
A066 014 L
A059 001 S
A096 014 L
A097 013 S
P004 008 L
A060 007 S
A068 012 L
A060 008 S
A097 011 L

A097 016 S
P004 011 L
A060 002 S
A068 014 L
A060 001 S
A097 014 L
A098 013 S
P003 016 L
A057 007 S
A070 012 L
A057 008 S
A098 011 L
A098 016 S
P003 011 L
A057 002 S
A070 014 L
A057 001 S
A098 014 L
A099 013 S
P005 016 L
A058 007 S
A072 012 L
A058 008 S
A099 011 L
A099 016 S
P005 007 L
A058 002 S
A072 014 L
A058 001 S
A099 014 L
P005 002 S
A066 013 L
A068 013 L
A070 013 L
A072 013 L
P004 003 S
A100 016 L
P004 007 S
A100 014 L
A100 002 S
A061 016 L
A074 007 S
A074 006 L
A074 001 S
A074 003 L
A077 007 S
A077 006 L
A077 001 S
A077 003 L
A075 007 S

A075 006 L
A075 001 S
A075 003 L
A076 007 S
A076 006 L
A076 001 S
A076 003 L
A096 007 S
A096 006 L
A096 001 S
A096 003 L
A098 007 S
A098 006 L
A098 001 S
A098 003 L
A097 007 S
A097 006 L
A097 001 S
A097 003 L
A099 007 S
A099 006 L
A099 001 S
A099 003 L
P013 004 S
A080 011 L
A088 016 L
P013 012 S
A080 016 L
A089 016 L
P014 008 S
A082 011 L
A090 016 L
P013 007 S
A082 016 L
A091 016 L
P014 012 S
A084 011 L
A092 016 L
P013 008 S
A084 016 L
A093 016 L
P014 015 S
A086 011 L
A094 016 L
P013 003 S
A086 016 L
A095 016 L
P015 003 S
A088 001 L
A089 001 L

A090 001 L
A091 001 L
A092 001 L
A093 001 L
A094 001 L
A095 001 L
P005 008 S
A064 002 L
A065 001 L
P004 012 S
A050 011 L
A050 016 L
A056 011 L
A056 016 L

FDQA

P005 004 S
A000 012 L
A001 012 L
A002 012 L
A003 012 L
A004 012 L
A005 012 L
A006 012 L
A007 012 L
P005 003 S
A000 014 L
A001 014 L
A002 014 L
A003 014 L
A004 014 L
A005 014 L
A006 014 L
A007 014 L
P005 015 S
A008 012 L
A009 012 L
A010 012 L
A011 012 L
A012 012 L
A013 012 L
A014 012 L
A015 012 L
P005 007 S
A008 014 L
A009 014 L
A010 014 L
A011 014 L
A012 014 L
A013 014 L
A014 014 L
A015 014 L
P006 007 S
A016 012 L
A017 012 L
A018 012 L
A019 012 L
A020 012 L
A021 012 L
A022 012 L
A023 012 L
P006 008 S
A016 014 L
A017 014 L
A018 014 L
A019 014 L

A020 014 L
A021 014 L
A022 014 L
A023 014 L
P006 004 S
A024 012 L
A025 012 L
A026 012 L
A027 012 L
A028 012 L
A029 012 L
A030 012 L
A031 012 L
P005 016 S
A024 014 L
A025 014 L
A026 014 L
A027 014 L
A028 014 L
A029 014 L
A030 014 L
A031 014 L
P004 003 S
A032 016 L
P004 007 S
A032 014 L
A032 002 S
A033 016 L
A034 001 S
A035 011 L
A035 014 L
A036 009 L
A036 014 L
A037 011 L
A037 014 L
A038 009 L
A038 014 L
A034 002 S
A039 011 L
A039 014 L
A040 009 L
A040 014 L
A041 011 L
A041 014 L
A042 009 L
A042 014 L
A033 001 S
A039 012 L
A039 013 L
A033 002 S

A040 012 L
A040 013 L
A041 012 L
A041 013 L
A033 003 S
A042 012 L
A042 013 L
A033 004 S
A035 012 L
A035 013 L
A033 005 S
A043 012 L
A043 013 L
A033 006 S
A038 012 L
A038 013 L
A044 014 L
A033 007 S
A036 012 L
A036 013 L
A037 012 L
A037 013 L
A033 008 S
A045 012 L
A045 013 L
A035 007 S
A000 011 L
A008 011 L
A035 005 S
A000 013 L
A008 013 L
A039 007 S
A016 011 L
A024 011 L
A039 005 S
A016 013 L
A024 013 L
A035 004 S
A001 011 L
A009 011 L
A035 002 S
A001 013 L
A009 013 L
A039 004 S
A017 011 L
A025 011 L
A039 002 S
A017 013 L
A025 013 L
A036 007 S

A002 011 L
A010 011 L
A036 005 S
A002 013 L
A010 013 L
A040 007 S
A018 011 L
A026 011 L
A040 005 S
A018 013 L
A026 013 L
A036 004 S
A003 011 L
A011 011 L
A036 002 S
A003 013 L
A011 013 L
A040 004 S
A019 011 L
A027 011 L
A040 002 S
A019 013 L
A027 013 L
A037 007 S
A004 011 L
A012 011 L
A037 005 S
A004 013 L
A012 013 L
A041 007 S
A020 011 L
A028 011 L
A041 005 S
A020 013 L
A028 013 L
A037 004 S
A005 011 L
A013 011 L
A037 002 S
A005 013 L
A013 013 L
A041 004 S
A021 011 L
A029 011 L
A041 002 S
A021 013 L
A029 013 L
A038 007 S
A006 011 L
A014 011 L

A038 005 S
A006 013 L
A014 013 L
A042 007 S
A022 011 L
A030 011 L
A042 005 S
A022 013 L
A030 013 L
A038 004 S
A007 011 L
A015 011 L
A038 002 S
A007 013 L
A015 013 L
A042 004 S
A023 011 L
A031 011 L
A042 002 S
A023 013 L
A031 013 L
A044 004 S
A046 011 L
A047 011 L
A044 002 S
A046 013 L
A047 013 L
A043 007 S
A048 011 L
A049 011 L
A043 005 S
A048 013 L
A049 013 L
A043 004 S
A050 011 L
A051 011 L
A043 002 S
A050 013 L
A051 013 L
A045 007 S
A052 011 L
A053 011 L
A045 005 S
A052 013 L
A053 013 L
A045 004 S
A054 011 L
A055 011 L
A045 002 S
A054 013 L

A055 013 L
P008 015 S
A034 013 L
A034 009 S
A045 009 L
A045 014 L
A034 008 S
A043 009 L
A043 014 L
A034 007 S
A044 016 L
A046 008 S
A056 012 L
A057 012 L
A058 012 L
A059 012 L
A046 007 S
A056 009 L
A056 014 L
A057 009 L
A057 014 L
A058 009 L
A058 014 L
A059 009 L
A059 014 L
A046 001 S
A060 012 L
A061 012 L
A062 012 L
A063 012 L
A046 002 S
A060 009 L
A060 014 L
A061 009 L
A061 014 L
A062 009 L
A062 014 L
A063 009 L
A063 014 L
A047 008 S
A064 012 L
A065 012 L
A066 012 L
A067 012 L
A047 007 S
A064 009 L
A064 014 L
A065 009 L
A065 014 L
A066 009 L

A066 014 L
A067 009 L
A067 014 L
A047 001 S
A068 012 L
A069 012 L
A070 012 L
A071 012 L
A047 002 S
A068 009 L
A068 014 L
A069 009 L
A069 014 L
A070 009 L
A070 014 L
A071 009 L
A071 014 L
P001 011 S
A072 014 L
A073 014 L
P002 007 S
A074 012 L
P002 011 S
A074 008 L
A074 005 S
A043 011 L
A043 016 L
A045 011 L
A045 016 L
A075 004 S
P007 004 L
A076 007 S
P007 007 L
A076 004 S
P007 003 L
A044 007 S
A044 001 L
A074 002 S
A044 008 L
A074 004 S
A077 011 L
A077 013 S
A046 012 L
A046 014 L
A048 012 L
A052 012 L
A047 012 L
A047 014 L
A072 005 S
A077 014 L

A077 016 S
A048 014 L
A052 014 L
A072 004 S
A078 011 L
A078 013 S
A049 012 L
A053 012 L
A072 001 S
A078 014 L
A078 016 S
A049 014 L
A053 014 L
A073 008 S
A079 011 L
A079 013 S
A050 012 L
A054 012 L
A073 005 S
A079 014 L
A079 016 S
A075 008 L
A050 014 L
A054 014 L
A075 007 S
A044 009 L
A073 004 S
A080 011 L
A080 013 S
A051 012 L
A055 012 L
A073 001 S
A080 014 L
A080 016 S
A051 014 L
A055 014 L
A081 005 S
A056 008 L
A056 016 L
A064 008 L
A064 016 L
A082 005 S
A057 008 L
A057 016 L
A065 008 L
A065 016 L
A083 005 S
A058 008 L
A058 016 L
A066 008 L

A066 016 L
A084 005 S
A059 008 L
A059 016 L
A067 008 L
A067 016 L
A085 005 S
A060 008 L
A060 016 L
A068 008 L
A068 016 L
A086 005 S
A061 008 L
A061 016 L
A069 008 L
A069 016 L
A087 005 S
A062 008 L
A062 016 L
A070 008 L
A070 016 L
A088 005 S
A063 008 L
A063 016 L
A071 008 L
A071 016 L
A056 005 S
P001 008 L
A056 007 S
P001 011 L
A056 004 S
P001 007 L
A056 002 S
P001 003 L
A064 005 S
P001 007 L
A064 007 S
P001 012 L
A064 004 S
P001 003 L
A064 002 S
P001 004 L
A057 005 S
P002 003 L
A057 007 S
P002 008 L
A057 004 S
P001 016 L
A057 002 S
P001 015 L

A065 005 S
P002 004 L
A065 007 S
P002 007 L
A065 004 S
P002 003 L
A065 002 S
P001 015 L
A058 005 S
P003 004 L
A058 007 S
P003 007 L
A058 004 S
P003 003 L
A058 002 S
P002 015 L
A066 005 S
P003 007 L
A066 007 S
P003 008 L
A066 004 S
P003 003 L
A066 002 S
P002 016 L
A059 005 S
P003 015 L
A059 007 S
P003 016 L
A059 004 S
P003 015 L
A059 002 S
P003 012 L
A067 005 S
P002 015 L
A067 007 S
P002 012 L
A067 004 S
P003 011 L
A067 002 S
P002 011 L
A060 005 S
P013 008 L
A060 007 S
P013 011 L
A060 004 S
P013 007 L
A060 002 S
P013 003 L
A068 005 S
P013 007 L

A068 007 S
P013 011 L
A068 004 S
P013 004 L
A068 002 S
P013 003 L
A061 005 S
P014 004 L
A061 007 S
P014 008 L
A061 004 S
P013 016 L
A061 002 S
P013 012 L
A069 005 S
P014 003 L
A069 007 S
P014 007 L
A069 004 S
P013 015 L
A069 002 S
P013 015 L
A062 005 S
P014 015 L
A062 007 S
P015 003 L
A062 004 S
P014 012 L
A062 002 S
P014 011 L
A070 005 S
P014 016 L
A070 007 S
P015 003 L
A070 004 S
P014 015 L
A070 002 S
P014 011 L
A063 005 S
P015 011 L
A063 007 S
P015 015 L
A063 004 S
P015 008 L
A063 002 S
P015 004 L
A071 005 S
P015 012 L
A071 007 S
P015 016 L

A071 004 S
P015 011 L
A071 002 S
P015 007 L
A089 005 S
A056 011 L
A056 013 L
A064 011 L
A064 013 L
A089 004 S
A057 011 L
A057 013 L
A065 011 L
A065 013 L
A090 005 S
A058 011 L
A058 013 L
A066 011 L
A066 013 L
A090 004 S
A059 011 L
A059 013 L
A067 011 L
A067 013 L
A091 005 S
A060 011 L
A060 013 L
A068 011 L
A068 013 L
A091 004 S
A061 011 L
A061 013 L
A069 011 L
A069 013 L
A092 005 S
A062 011 L
A062 013 L
A070 011 L
A070 013 L
A092 004 S
A063 011 L
A063 013 L
A071 011 L
A071 013 L
A000 007 S
A081 002 L
A000 002 S
A082 007 L
A008 007 S
A083 007 L

A008 002 S
A084 007 L
A016 007 S
A085 007 L
A016 002 S
A086 007 L
A024 007 S
A087 007 L
A024 002 S
A088 007 L
A001 007 S
A081 009 L
A001 002 S
A082 009 L
A009 007 S
A083 009 L
A009 002 S
A084 009 L
A017 007 S
A085 009 L
A017 002 S
A086 009 L
A025 007 S
A087 009 L
A025 002 S
A088 009 L
A002 007 S
A081 012 L
A002 002 S
A082 012 L
A010 007 S
A083 012 L
A010 002 S
A084 012 L
A018 007 S
A085 012 L
A018 002 S
A086 012 L
A026 007 S
A087 012 L
A026 002 S
A088 012 L
A003 007 S
A081 014 L
A003 002 S
A082 014 L
A011 007 S
A083 014 L
A011 002 S
A084 014 L

A019 007 S
A085 014 L
A019 002 S
A086 014 L
A027 007 S
A087 014 L
A027 002 S
A088 014 L
A004 007 S
A081 001 L
A004 002 S
A082 001 L
A012 007 S
A083 001 L
A012 002 S
A084 001 L
A020 007 S
A085 001 L
A020 002 S
A086 001 L
A028 007 S
A087 001 L
A028 002 S
A088 001 L
A005 007 S
A089 008 L
A005 002 S
A089 001 L
A013 007 S
A090 008 L
A013 002 S
A090 001 L
A021 007 S
A091 008 L
A021 002 S
A091 001 L
A029 007 S
A092 008 L
A029 002 S
A092 001 L
A006 007 S
A089 009 L
A006 002 S
A089 016 L
A014 007 S
A090 009 L
A014 002 S
A090 016 L
A022 007 S
A091 009 L

A022 002 S
A091 016 L
A030 007 S
A092 009 L
A030 002 S
A092 016 L
A007 007 S
A089 012 L
A077 002 S
A089 014 L
A015 007 S
A090 012 L
A015 002 S
A090 014 L
A023 007 S
A091 012 L
A023 002 S
A091 014 L
A031 007 S
A092 012 L
A031 002 S
A092 014 L
P006 016 S
A035 008 L
A039 008 L
P005 008 S
A035 001 L
A039 001 L
P006 012 S
A036 011 L
A040 011 L
P005 011 S
A036 001 L
A040 001 L
P006 015 S
A037 008 L
A041 008 L
P005 012 S
A037 001 L
A041 001 L
P006 011 S
A038 011 L
A042 011 L
P006 003 S
A038 001 L
A042 001 L
A048 007 S
A081 008 L
A082 008 L
A083 008 L

A084 008 L
A072 012 L
A052 007 S
A085 008 L
A086 008 L
A087 008 L
A088 008 L
A048 001 S
A075 013 L
A048 002 S
A081 011 L
A082 011 L
A083 011 L
A084 011 L
A072 013 L
A052 002 S
A085 011 L
A086 011 L
A087 011 L
A088 011 L
A049 008 S
A076 011 L
A049 007 S
A081 013 L
A082 013 L
A083 013 L
A084 013 L
A072 016 L
A053 007 S
A085 013 L
A086 013 L
A087 013 L
A088 013 L
A049 001 S
A075 001 L
A076 008 L
A049 002 S
A081 016 L
A082 016 L
A083 016 L
A084 016 L
A073 009 L
A053 002 S
A085 016 L
A086 016 L
A087 016 L
A088 016 L
A050 008 S
A076 001 L
A050 007 S

A081 002 L
A082 002 L
A083 002 L
A084 002 L
A073 012 L
A054 007 S
A085 002 L
A086 002 L
A087 002 L
A088 002 L
A050 001 S
A075 014 L
A076 016 L
A050 002 S
A089 007 L
A090 007 L
A073 013 L
A054 002 S
A091 007 L
A092 007 L
A051 008 S
A076 012 L
A076 013 L
A051 007 S
A089 011 L
A090 011 L
A073 016 L
A055 007 S
A091 011 L
A092 011 L
A051 001 S
A075 016 L
A076 009 L
A076 014 L
A051 002 S
A089 013 L
A090 013 L
A074 014 L
A055 002 S
A091 013 L
A092 013 L
A077 007 S
A077 006 L
A077 001 S
A077 003 L
A078 007 S
A078 006 L
A078 001 S
A078 003 L
A079 007 S
A079 006 L
A079 001 S
A079 003 L
A080 007 S
A080 006 L
A080 001 S
A080 003 L
P008 011 S
A034 011 L
P008 007 S
A034 014 L

FIRDB

P004 003 S
A000 016 L
P004 007 S
A000 014 L
A000 002 S
A001 016 L
A002 008 S
A003 009 L
A003 016 L
A004 001 L
A004 009 L
A005 009 L
A005 014 L
A003 004 S
A006 011 L
A007 011 L
A003 005 S
A006 013 L
A007 013 L
A005 004 S
A008 011 L
A009 011 L
A005 005 S
A008 013 L
A009 013 L
A004 004 S
A010 011 L
A004 005 S
A010 013 L
A001 001 S
A003 012 L
A003 013 L
A005 012 L
A005 013 L
A001 002 S
A004 012 L
A004 013 L
P007 007 S
A002 011 L
A011 002 S
A003 008 L
A004 011 L
A005 008 L
A012 007 S
A003 001 L
A004 014 L
A005 001 L
P007 004 S
A002 012 L
P006 016 S

A012 013 L
A013 004 S
A011 012 L
P011 011 S
A014 009 L
A014 008 S
A006 012 L
P011 007 S
A014 012 L
A014 005 S
A006 014 L
P011 003 S
A014 013 L
A014 004 S
A007 012 L
P010 015 S
A014 016 L
A014 001 S
A007 014 L
P009 015 S
A015 009 L
A015 008 S
A008 012 L
P009 016 S
A015 012 L
A015 005 S
A008 014 L
P009 011 S
A015 013 L
A015 004 S
A009 012 L
P009 012 S
A015 016 L
A015 001 S
A009 014 L
P007 003 S
A002 016 L
A002 002 S
A010 012 L
A002 005 S
A013 013 L
A012 011 L
A012 004 S
A013 012 L
A012 012 L
P006 015 S
A012 008 L
P009 003 S
A011 008 L
P008 015 S

A011 011 L
A016 005 S
A017 007 L
A018 007 L
A019 007 L
A016 007 S
A020 007 L
A021 007 L
A016 008 S
A022 007 L
A023 007 L
A024 007 L
A025 007 L
A026 007 L
A016 009 S
P014 007 L
A006 007 S
A016 011 L
A027 007 L
A028 005 S
A029 007 L
A030 007 L
A031 007 L
A032 007 L
A033 007 L
A028 007 S
A017 008 L
A018 008 L
A019 008 L
A020 008 L
A021 008 L
A028 001 S
A034 007 L
A035 007 L
A036 007 L
A037 007 L
A038 007 L
A039 007 L
A040 007 L
A028 008 S
A022 008 L
A023 008 L
A024 008 L
A025 008 L
A026 008 L
A028 002 S
A041 007 L
A042 007 L
A028 009 S
A043 007 L

A044 007 L
A045 008 L
A045 013 L
A028 004 S
P013 007 L
A006 001 S
A028 014 L
A027 008 L
A046 007 L
A006 002 S
A028 011 L
A016 004 S
A046 008 L
A032 008 L
A033 008 L
A038 008 L
A040 008 L
A023 009 L
A025 009 L
A026 009 L
A016 001 S
A027 009 L
A029 008 L
A030 008 L
A031 008 L
A017 009 L
A016 002 S
A018 009 L
A019 009 L
A020 009 L
A021 009 L
A034 008 L
A035 008 L
A036 008 L
A037 008 L
A016 016 S
A039 008 L
A022 009 L
A024 009 L
A042 008 L
A043 008 L
A044 008 L
A045 009 L
A045 016 L
A007 008 S
A041 008 L
P006 011 L
A007 007 S
A016 014 L
A047 005 S

A027 011 L
A029 009 L
A047 007 S
A046 009 L
A030 009 L
A031 009 L
A032 009 L
A033 009 L
A017 011 L
A047 001 S
A038 009 L
A039 009 L
A040 009 L
A023 011 L
A024 011 L
A025 011 L
A026 011 L
A047 008 S
A018 011 L
A019 011 L
A020 011 L
A021 011 L
A034 009 L
A035 009 L
A036 009 L
A037 009 L
A047 002 S
A041 009 L
A042 009 L
A043 009 L
A044 009 L
A047 009 S
A022 011 L
A047 004 S
P011 008 L
A007 001 S
A047 014 L
A007 002 S
A047 011 L
A048 005 S
A046 011 L
A029 011 L
A034 011 L
A035 011 L
A036 011 L
A037 011 L
A043 011 L
A045 012 L
A048 007 S
A027 012 L

A030 011 L
A031 011 L
A032 011 L
A033 011 L
A017 012 L
A048 008 S
A018 012 L
A019 012 L
A020 012 L
A021 012 L
A041 011 L
A042 011 L
A048 009 S
A038 011 L
A039 011 L
A040 011 L
A022 012 L
A023 012 L
A024 012 L
A025 012 L
A026 012 L
A008 008 S
P009 016 L
A008 007 S
A048 011 L
A049 005 S
A027 013 L
A030 012 L
A031 012 L
A032 012 L
A033 012 L
A017 013 L
A049 007 S
A046 012 L
A029 012 L
A022 013 L
A023 013 L
A024 013 L
A025 013 L
A026 013 L
A049 001 S
A018 013 L
A019 013 L
A020 013 L
A021 013 L
A034 012 L
A035 012 L
A036 012 L
A037 012 L
A049 008 S

A041 012 L
A042 012 L
A049 002 S
A038 012 L
A039 012 L
A040 012 L
A044 011 L
A045 001 L
A049 009 S
P009 004 L
A008 001 S
A049 014 L
A008 002 S
A049 011 L
A050 005 S
A046 013 L
A029 013 L
A017 014 L
A018 014 L
A019 014 L
A020 014 L
A021 014 L
A050 007 S
A027 014 L
A030 013 L
A031 013 L
A032 013 L
A033 013 L
A050 001 S
A023 014 L
A041 013 L
A042 013 L
A050 008 S
A034 013 L
A035 013 L
A036 013 L
A037 013 L
A043 012 L
A044 012 L
A045 011 L
A045 014 L
A050 009 S
A038 013 L
A039 013 L
A040 013 L
A022 014 L
A024 014 L
A025 014 L
A026 014 L
A050 004 S

P007 004 L
A009 008 S
A050 014 L
A009 007 S
A050 011 L
A051 005 S
A029 014 L
A030 014 L
A032 014 L
A051 007 S
A027 016 L
A046 014 L
A031 014 L
A033 014 L
A017 016 L
A020 016 L
A034 014 L
A036 014 L
A051 001 S
A018 016 L
A019 016 L
A021 016 L
A035 014 L
A037 014 L
A051 008 S
A038 014 L
A039 014 L
A040 014 L
A022 016 L
A026 016 L
A041 014 L
A042 014 L
A051 002 S
A023 016 L
A024 016 L
A025 016 L
A051 004 S
P006 007 L
A009 001 S
A051 014 L
A009 002 S
A051 011 L
A048 004 S
A017 001 L
A019 001 L
A020 001 L
A036 016 L
A037 016 L
A039 016 L
A040 016 L

A048 001 S
A027 001 L
A046 016 L
A029 016 L
A030 016 L
A031 016 L
A032 016 L
A033 016 L
A048 002 S
A018 001 L
A021 001 L
A034 016 L
A035 016 L
A038 016 L
A048 016 S
P010 007 L
A010 007 S
A048 014 L
P008 016 S
A011 009 L
P006 011 S
A012 009 L
P008 007 S
A011 001 L
A037 002 S
P015 004 L
A036 002 S
P001 008 L
A031 002 S
P010 016 L
A030 002 S
P012 011 L
A033 002 S
P009 007 L
A032 002 S
P010 004 L
A029 002 S
P013 003 L
A040 002 S
P012 007 L
A042 002 S
P002 016 L
A041 002 S
P006 012 L
A038 002 S
P013 015 L
A046 002 S
P013 011 L
A039 002 S
P013 004 L

A035 002 S
P003 003 L
A034 002 S
P003 015 L
A013 001 S
P002 015 L
A043 004 S
A013 016 L
A044 004 S
A013 014 L
A045 002 S
P001 007 L
A022 002 S
P011 004 L
A024 002 S
P009 015 L
A025 002 S
P008 008 L
A026 002 S
P007 003 L
A023 002 S
P010 003 L
A021 002 S
P004 004 L
A020 002 S
P005 011 L
A027 002 S
P015 005 L
A019 002 S
P006 015 L
A018 002 S
P007 008 L
A017 002 S
P008 003 L

IICRI

P004 015 S
A000 013 L
P004 011 S
A001 013 L
P002 007 S
A002 013 L
P002 011 S
A003 013 L
P001 003 S
A004 013 L
P001 004 S
A005 013 L
P004 011 S
A006 013 L
P003 011 S
A007 013 L
P005 015 S
A008 013 L
P005 012 S
A009 013 L
P005 007 S
A010 013 L
P013 012 S
A011 013 L
P014 008 S
A012 014 L
A013 012 L
P015 011 S
A014 008 L
A015 008 L
P015 007 S
A014 014 L
A015 014 L
P014 003 S
A016 008 L
A017 008 L
P014 003 S
A016 014 L
A017 014 L
P013 015 S
A018 008 L
A019 008 L
P013 015 S
A018 014 L
A019 014 L
P014 007 S
A020 008 L
A021 008 L
P013 012 S
A020 014 L

A021 014 L
P013 003 S
A022 008 L
A023 008 L
P013 004 S
A022 014 L
A023 014 L
P013 016 S
A024 008 L
A011 008 L
A025 007 S
A015 009 L
A015 016 L
A017 009 L
A017 016 L
A019 009 L
A019 016 L
A021 009 L
A021 016 L
A025 008 S
A023 009 L
A023 016 L
A026 009 L
A013 011 L
A011 009 L
P005 008 S
A027 013 L
A027 001 S
A014 009 L
A014 016 L
A016 009 L
A016 016 L
A018 009 L
A018 016 L
A020 009 L
A020 016 L
A027 002 S
A022 009 L
A022 016 L
A012 009 L
A012 016 L
A024 009 L
P007 012 S
A028 013 L
A028 001 S
A000 014 L
A001 014 L
A002 014 L
A003 014 L
A004 014 L

A005 014 L
A006 014 L
A007 014 L
A028 002 S
A008 014 L
A009 014 L
A029 014 L
A030 014 L
A010 014 L
P004 003 S
A031 016 L
P004 007 S
A031 014 L
A031 002 S
A032 016 L
P004 015 S
A025 011 L
A032 001 S
A033 012 L
A033 013 L
A034 012 L
A032 002 S
A035 012 L
A035 013 L
A032 003 S
A036 012 L
A036 013 L
A037 012 L
A037 013 L
A032 004 S
A038 012 L
A038 013 L
A032 005 S
A039 012 L
A039 013 L
A040 012 L
A040 013 L
A032 006 S
A041 012 L
A041 013 L
A032 007 S
A042 012 L
A042 013 L
A032 008 S
A043 012 L
A043 013 L
P008 003 S
A044 016 L
A044 001 S
A038 009 L

A038 016 L
A041 009 L
A041 016 L
A043 009 L
A043 016 L
A037 009 L
A037 016 L
P009 016 S
A034 008 L
A038 007 S
A045 011 L
A046 011 L
A038 005 S
A045 013 L
A046 013 L
A038 004 S
A047 011 L
A048 011 L
A038 002 S
A047 013 L
A048 013 L
A041 004 S
A049 011 L
A050 011 L
A041 002 S
A049 013 L
A050 013 L
A041 007 S
A051 011 L
A052 011 L
A041 005 S
A051 013 L
A052 013 L
A043 004 S
A053 011 L
A054 011 L
A043 002 S
A053 013 L
A054 013 L
A043 007 S
A055 011 L
A056 011 L
A043 005 S
A055 013 L
A056 013 L
A037 007 S
A057 011 L
A058 011 L
A037 005 S
A057 013 L

A058 013 L
A037 004 S
A059 011 L
A060 011 L
A037 002 S
A059 013 L
A060 013 L
A034 007 S
A061 011 L
A034 005 S
A061 013 L
A062 007 S
A039 011 L
A039 014 L
A036 011 L
A036 014 L
A042 011 L
A042 014 L
A033 011 L
A033 014 L
A062 008 S
A040 011 L
A040 014 L
A035 011 L
A035 014 L
A038 011 L
A038 014 L
A041 011 L
A041 014 L
A062 009 S
A043 011 L
A043 014 L
A037 011 L
A037 014 L
A034 011 L
A063 001 S
A064 012 L
A065 012 L
A066 012 L
A067 012 L
A068 012 L
A069 012 L
A070 012 L
A071 012 L
A063 002 S
A072 012 L
A073 012 L
A074 012 L
A075 012 L
A076 012 L

P010 008 S
A063 013 L
A044 004 S
A040 009 L
A040 016 L
A035 009 L
P008 007 S
A044 013 L
A040 004 S
A077 011 L
A078 011 L
A040 002 S
A077 013 L
A078 013 L
A040 007 S
A079 011 L
A080 011 L
A040 005 S
A079 013 L
A080 013 L
A035 007 S
A081 011 L
A082 011 L
A035 005 S
A081 013 L
A082 013 L
A035 004 S
A083 011 L
A084 011 L
A035 002 S
A083 013 L
A084 013 L
P009 015 S
A035 016 L
A063 007 S
A064 009 L
A065 009 L
A066 009 L
A067 009 L
A068 009 L
A069 009 L
A070 009 L
A071 009 L
A063 008 S
A072 009 L
A073 009 L
A074 009 L
A075 009 L
A076 009 L
P010 011 S

A063 011 L
A044 005 S
A042 009 L
A042 016 L
A033 009 L
A033 016 L
P008 008 S
A044 012 L
A042 004 S
A085 011 L
A086 011 L
A042 002 S
A085 013 L
A086 013 L
A042 007 S
A087 011 L
A088 011 L
A042 005 S
A087 013 L
A088 013 L
A033 007 S
A089 011 L
A090 011 L
A033 005 S
A089 013 L
A090 013 L
A033 004 S
A091 011 L
A033 002 S
A091 013 L
A092 001 S
A064 001 L
A065 001 L
A066 001 L
A067 001 L
A068 001 L
A069 001 L
A070 001 L
A071 001 L
A092 002 S
A072 001 L
A073 001 L
A074 001 L
A075 001 L
A076 001 L
P006 004 S
A092 013 L
A092 007 S
A064 014 L
A065 014 L

A066 014 L
A067 014 L
A068 014 L
A069 014 L
A070 014 L
A071 014 L
A092 008 S
A072 014 L
A073 014 L
A074 014 L
A075 014 L
A076 014 L
P006 007 S
A092 011 L
A044 008 S
A039 009 L
A039 016 L
A036 009 L
A036 016 L
P008 011 S
A044 009 L
A039 004 S
A093 011 L
A094 011 L
A039 002 S
A093 013 L
A094 013 L
A039 007 S
A095 011 L
A096 011 L
A039 005 S
A095 013 L
A096 013 L
A036 007 S
A097 011 L
A098 011 L
A036 005 S
A097 013 L
A098 013 L
A036 004 S
A099 011 L
A036 002 S
A099 013 L
A034 002 S
P009 012 L
A034 004 S
P010 014 L
A100 005 S
A101 007 L
A102 005 S

A101 008 L
A103 005 S
A101 009 L
A104 005 S
A101 011 L
A105 005 S
A101 012 L
A106 005 S
A101 013 L
A107 007 S
A101 014 L
P003 012 S
A108 013 L
P015 011 S
A000 016 L
A109 009 L
P015 008 S
A001 016 L
A109 016 L
P010 016 S
A002 016 L
A110 009 L
P015 015 S
A003 016 L
A110 016 L
P015 015 S
A004 016 L
A111 009 L
P009 003 S
A005 016 L
A111 016 L
P003 007 S
A006 016 L
A112 009 L
P003 008 S
A007 016 L
A112 016 L
P003 004 S
A008 016 L
A113 009 L
P002 016 S
A009 016 L
A113 016 L
P012 016 S
A010 016 L
P014 004 S
A030 016 L
P012 003 S
A114 013 L
A114 001 S

A109 011 L
A110 011 L
A111 011 L
A112 011 L
A114 002 S
A113 011 L
A115 011 L
P007 011 S
A028 011 L
A028 007 S
A000 001 L
A001 001 L
A002 001 L
A003 001 L
A004 001 L
A005 001 L
A006 001 L
A007 001 L
A028 008 S
A008 001 L
A009 001 L
A029 001 L
A030 001 L
A010 001 L
A050 008 S
P004 014 L
A050 007 S
P002 012 L
A061 008 S
P011 010 L
A061 007 S
P006 003 L
A061 001 S
P011 003 L
A061 002 S
P006 002 L
A045 008 S
P008 015 L
A045 007 S
P001 012 L
A051 008 S
P010 003 L
A051 007 S
P005 003 L
A051 001 S
P010 004 L
A051 002 S
P004 016 L
A045 001 S
P008 003 L

A045 002 S
P001 011 L
A052 008 S
P010 007 L
A052 007 S
P005 004 L
A052 001 S
P010 011 L
A052 002 S
P005 003 L
A046 008 S
P009 011 L
A046 007 S
P001 016 L
A053 008 S
P011 015 L
A053 007 S
P005 011 L
A053 001 S
P011 011 L
A053 002 S
P006 007 L
A046 001 S
P009 007 L
A046 002 S
P001 015 L
A054 008 S
P011 003 L
A054 007 S
P004 004 L
A054 001 S
P011 006 L
A054 002 S
P004 008 L
A047 008 S
P009 007 L
A047 007 S
P001 008 L
A055 008 S
P011 016 L
A055 007 S
P005 016 L
A055 001 S
P011 015 L
A055 002 S
P006 008 L
A047 001 S
P008 016 L
A047 002 S
P001 007 L

A056 008 S
P011 011 L
A056 007 S
P004 007 L
A056 001 S
P011 012 L
A056 002 S
P004 003 L
A048 008 S
P008 015 L
A048 007 S
P001 011 L
A057 008 S
P012 011 L
A057 007 S
P006 016 L
A057 001 S
P012 012 L
A057 002 S
P006 015 L
A048 001 S
P008 012 L
A048 002 S
P001 012 L
A058 008 S
P011 007 L
A058 007 S
P006 003 L
A058 001 S
P011 008 L
A058 002 S
P006 012 L
A049 008 S
P007 004 L
A049 007 S
P003 015 L
A059 008 S
P012 015 L
A059 007 S
P006 006 L
A059 001 S
P012 008 L
A059 002 S
P006 011 L
A049 001 S
P007 003 L
A049 002 S
P003 016 L
A060 008 S
P012 007 L

A060 007 S
P006 015 L
A060 001 S
P012 003 L
A060 002 S
P006 011 L
A116 004 S
P015 004 L
A117 008 S
P014 016 L
A117 005 S
P013 016 L
A117 004 S
P011 002 L
A117 001 S
P010 015 L
A118 008 S
P015 003 L
A118 005 S
P015 003 L
A118 004 S
P003 004 L
A118 001 S
P003 003 L
A116 008 S
P002 004 L
A116 005 S
P002 003 L
A116 001 S
P010 015 L
A119 007 S
P015 016 L
P012 004 S
A114 011 L
A082 008 S
A116 013 L
A115 012 L
A074 011 L
A106 008 L
A082 007 S
A012 011 L
A074 008 L
A106 011 L
A077 008 S
A109 012 L
A064 011 L
A100 008 L
A117 009 L
A077 007 S
A014 011 L

A064 008 L
A100 011 L
A077 001 S
A109 014 L
A065 011 L
A100 013 L
A117 012 L
A077 002 S
A014 013 L
A065 008 L
A100 016 L
A078 008 S
A117 013 L
A110 012 L
A066 011 L
A102 008 L
A078 007 S
A016 011 L
A066 008 L
A102 011 L
A078 001 S
A117 016 L
A110 014 L
A067 011 L
A102 013 L
A078 002 S
A016 013 L
A067 008 L
A102 016 L
A079 008 S
A118 009 L
A111 012 L
A068 011 L
A103 008 L
A079 007 S
A018 011 L
A068 008 L
A103 011 L
A079 001 S
A118 012 L
A111 014 L
A069 011 L
A103 013 L
A079 002 S
A018 013 L
A069 008 L
A103 016 L
A080 008 S
A118 013 L
A112 012 L

A070 011 L
A104 008 L
A080 007 S
A020 011 L
A070 008 L
A104 011 L
A080 001 S
A118 016 L
A112 014 L
A071 011 L
A104 013 L
A080 002 S
A020 013 L
A071 008 L
A104 016 L
A081 008 S
A116 009 L
A113 012 L
A072 011 L
A105 008 L
A081 007 S
A022 011 L
A072 008 L
A105 011 L
A081 001 S
A116 012 L
A113 014 L
A073 011 L
A105 013 L
A081 002 S
A022 013 L
A073 008 L
A105 016 L
A119 001 S
P015 008 L
A119 004 S
P015 012 L
A084 008 S
A116 016 L
A119 013 L
A076 011 L
A107 008 L
A084 007 S
A024 011 L
A076 008 L
A107 011 L
A084 001 S
P009 016 L
A084 002 S
P009 015 L

A119 008 S
P015 016 L
A119 005 S
P015 012 L
A083 008 S
A075 011 L
A106 013 L
A119 011 L
A083 007 S
A012 013 L
A075 008 L
A106 016 L
A083 001 S
P012 007 L
A083 002 S
P012 011 L
A062 001 S
A109 013 L
A110 013 L
A111 013 L
A112 013 L
A062 002 S
A113 013 L
A115 013 L
P011 004 S
A062 013 L
P005 007 S
A027 011 L
A027 007 S
A014 012 L
A016 012 L
A018 012 L
A020 012 L
A027 008 S
A022 012 L
A012 012 L
A024 012 L
A026 007 S
A029 008 L
A015 007 S
A000 008 L
A015 002 S
A001 008 L
A017 007 S
A002 008 L
A017 002 S
A003 008 L
A019 007 S
A004 008 L
A019 002 S

A005 008 L
A021 007 S
A006 008 L
A021 002 S
A007 008 L
A023 007 S
A008 008 L
A023 002 S
A009 008 L
A011 005 S
A010 008 L
A013 005 S
A030 008 L
P009 011 S
A120 011 L
A120 007 S
A000 009 L
A001 009 L
A002 009 L
A003 009 L
A004 009 L
A005 009 L
A006 009 L
A007 009 L
A120 008 S
A008 009 L
A009 009 L
A029 009 L
A030 009 L
A010 009 L
A029 005 S
A082 012 L
A000 005 S
A077 012 L
A001 005 S
A077 014 L
A002 005 S
A078 012 L
A003 005 S
A078 014 L
A004 005 S
A079 012 L
A005 005 S
A079 014 L
A006 005 S
A080 012 L
A007 005 S
A080 014 L
A008 005 S
A081 012 L

A009 005 S
A081 014 L
A010 005 S
A084 012 L
A084 014 L
A030 005 S
A083 012 L
A083 014 L
A074 005 S
P010 003 L
A074 007 S
P013 007 L
A064 005 S
P005 011 L
A064 007 S
P015 007 L
A065 005 S
P005 015 L
A065 007 S
P014 012 L
A066 005 S
P008 004 L
A066 007 S
P014 015 L
A067 005 S
P005 002 L
A067 007 S
P015 004 L
A068 005 S
P007 012 L
A068 007 S
P014 011 L
A069 005 S
P007 016 L
A069 007 S
P014 011 L
A070 005 S
P008 012 L
A070 007 S
P013 011 L
A071 005 S
P009 004 L
A071 007 S
P013 008 L
A072 005 S
P008 008 L
A072 007 S
P013 011 L
A073 005 S
P009 008 L

A073 007 S
P013 007 L
A076 005 S
P008 016 L
A076 007 S
P013 003 L
A075 005 S
P010 007 L
A075 007 S
P014 007 L
A012 005 S
A090 012 L
A014 005 S
A085 012 L
A014 004 S
A085 014 L
A016 005 S
A086 012 L
A016 004 S
A086 014 L
A018 005 S
A087 012 L
A018 004 S
A087 014 L
A020 005 S
A088 012 L
A020 004 S
A088 014 L
A022 005 S
A089 012 L
A022 004 S
A089 014 L
A024 005 S
A091 012 L
A012 004 S
A090 014 L
A115 005 S
A050 012 L
A061 012 L
A061 014 L
A109 005 S
A045 012 L
A051 012 L
A051 014 L
A109 004 S
A045 014 L
A052 012 L
A052 014 L
A110 005 S
A046 012 L

A053 012 L
A053 014 L
A110 004 S
A046 014 L
A054 012 L
A054 014 L
A111 005 S
A047 012 L
A055 012 L
A055 014 L
A111 004 S
A047 014 L
A056 012 L
A056 014 L
A112 005 S
A048 012 L
A057 012 L
A057 014 L
A112 004 S
A048 014 L
A058 012 L
A058 014 L
A113 005 S
A049 012 L
A059 012 L
A059 014 L
A113 004 S
A049 014 L
A060 012 L
A060 014 L
A090 008 S
A029 011 L
A074 016 L
A090 007 S
A074 013 L
A085 008 S
A000 011 L
A064 016 L
A085 007 S
A064 013 L
A085 001 S
A001 011 L
A065 016 L
A085 002 S
A065 013 L
A086 008 S
A002 011 L
A066 016 L
A086 007 S
A066 013 L

A086 001 S
A003 012 L
A067 016 L
A086 002 S
A067 013 L
A087 008 S
A004 011 L
A068 016 L
A087 007 S
A068 013 L
A087 001 S
A005 011 L
A069 016 L
A087 002 S
A069 013 L
A088 008 S
A006 011 L
A070 016 L
A088 007 S
A070 013 L
A088 001 S
A007 011 L
A071 016 L
A088 002 S
A071 013 L
A089 008 S
A008 011 L
A072 016 L
A089 007 S
A072 013 L
A089 001 S
A009 011 L
A073 016 L
A089 002 S
A073 013 L
A091 008 S
A010 011 L
A076 016 L
A091 007 S
A076 013 L
A090 001 S
A030 011 L
A075 016 L
A090 002 S
A075 013 L
A120 001 S
A000 012 L
A001 012 L
A002 012 L
A003 011 L

A004 012 L
A005 012 L
A006 012 L
A007 012 L
A120 002 S
A008 012 L
A009 012 L
A029 012 L
A030 012 L
A010 012 L
P009 004 S
A120 013 L
P003 012 S
A108 011 L
A108 007 S
A011 014 L
A101 002 S
P008 004 L
A098 008 S
A106 012 L
A098 007 S
A106 009 L
P010 012 L
A093 008 S
A100 012 L
A093 007 S
A100 009 L
P007 007 L
A093 001 S
A100 001 L
A093 002 S
A100 014 L
P007 008 L
A094 008 S
A102 012 L
A094 007 S
A102 009 L
P007 007 L
A094 001 S
A102 001 L
A094 002 S
A102 014 L
P007 003 L
A095 008 S
A103 012 L
A095 007 S
A103 009 L
P007 015 L
A095 001 S
A103 001 L

A095 002 S
A103 014 L
P007 011 L
A096 008 S
A104 012 L
A096 007 S
A104 009 L
P008 011 L
A096 001 S
A104 001 L
A096 002 S
A104 014 L
P007 016 L
A097 008 S
A105 012 L
A097 007 S
A105 009 L
P009 012 L
A097 001 S
A105 001 L
A097 002 S
A105 014 L
P009 003 L
A099 008 S
A107 012 L
A099 007 S
A107 009 L
P009 008 L
A098 001 S
A106 001 L
A098 002 S
A106 014 L
P008 007 L
P013 004 S
A011 001 L
P013 008 S
A013 014 L
A025 001 S
A013 013 L
A011 016 L
P004 012 S
A025 013 L
P014 015 S
A012 008 L
A026 008 L
P014 016 S
A026 011 L
A098 012 L
P014 012 S
A030 013 L

P012 015 S
A115 008 L
P014 008 S
A029 013 L
P014 004 S
A029 016 L
A115 009 L
P011 007 S
A062 011 L
P002 015 S
A109 008 L
P002 015 S
A109 001 L
P002 011 S
A110 008 L
P002 008 S
A110 001 L
P001 008 S
A111 008 L
P001 003 S
A111 001 L
P003 016 S
A112 008 L
P003 008 S
A112 001 L
P001 016 S
A113 008 L
P001 015 S
A113 001 L
A114 007 S
A109 007 L
A110 007 L
A111 007 L
A112 007 L
A114 008 L
A113 007 L
A115 007 L
P002 016 S
A015 011 L
A093 012 L
P003 003 S
A015 013 L
A093 014 L
P002 012 S
A017 011 L
A094 012 L
P002 008 S
A017 013 L
A094 014 L
P001 007 S

A019 011 L
A095 012 L
P001 004 S
A019 013 L
A095 014 L
P003 015 S
A021 011 L
A096 012 L
P003 011 S
A021 013 L
A096 014 L
P002 007 S
A023 011 L
A097 012 L
P002 004 S
A023 013 L
A097 014 L
P002 003 S
A011 011 L
A099 012 L
P003 007 S
A098 014 L
A013 009 L
A108 001 S
A015 012 L
A017 012 L
A019 012 L
A021 012 L
A108 002 S
A023 012 L
A026 012 L
A013 008 L
A011 012 L

P004 003 S
A000 016 L
P004 007 S
A000 014 L
A001 001 S
A002 011 L
A001 002 S
A002 016 L
A001 003 S
A003 011 L
A001 004 S
A003 016 L
A001 005 S
A004 011 L
A001 006 S
A004 016 L
A001 007 S
A005 011 L
A001 008 S
A005 016 L
A006 002 S
P014 008 L
P014 003 S
A007 011 L
P015 015 S
A008 009 L
P010 016 S
A009 011 L
P010 015 S
A010 011 L
P010 011 S
A011 011 L
P012 003 S
A012 013 L
P013 003 S
A013 009 L
P013 003 S
A014 009 L
P012 007 S
A014 016 L
A004 007 S
A015 011 L
A016 011 L
A004 005 S
A015 013 L
A016 013 L
A005 007 S
A017 011 L
A018 011 L
A005 005 S

A017 013 L
A018 013 L
A003 007 S
A019 011 L
A020 011 L
A003 005 S
A019 013 L
A020 013 L
A002 007 S
A021 011 L
A022 011 L
A002 005 S
A021 013 L
A022 013 L
P014 004 S
A007 014 L
A004 004 S
A023 011 L
A024 011 L
A004 002 S
A023 013 L
A024 013 L
A005 004 S
A025 011 L
A026 011 L
A005 002 S
A025 013 L
A026 013 L
A003 004 S
A027 011 L
A028 011 L
A003 002 S
A027 013 L
A028 013 L
A002 004 S
A029 011 L
A030 011 L
A002 002 S
A029 013 L
A030 013 L
A007 007 S
A004 001 L
A004 012 L
A005 001 L
A005 012 L
A003 001 L
A003 012 L
A002 001 L
A002 012 L
A031 005 S

P004 008 L
A032 002 S
P003 006 L
A023 007 S
A032 013 L
A031 012 L
A031 004 S
P004 004 L
A032 007 S
P003 006 L
A023 001 S
A006 014 L
A023 002 S
A032 012 L
A031 013 L
A033 005 S
P004 015 L
A034 002 S
P003 010 L
A024 007 S
A034 013 L
A033 012 L
A006 013 L
A033 004 S
P004 015 L
A034 007 S
P003 010 L
A024 002 S
A034 012 L
A033 013 L
A006 012 L
A031 008 S
P005 004 L
A035 002 S
P001 006 L
A015 008 S
A006 011 L
A015 007 S
A031 009 L
A035 013 L
A031 001 S
P004 007 L
A035 007 S
P001 006 L
A015 002 S
A031 016 L
A035 012 L
A033 008 S
P005 003 L
A036 002 S

P001 012 L
A016 007 S
A036 013 L
A033 009 L
A033 001 S
P004 012 L
A036 007 S
P001 012 L
A016 002 S
A036 012 L
A033 016 L
A037 008 S
P004 016 L
A038 002 S
P001 008 L
A025 008 S
A037 012 L
A025 007 S
A037 009 L
A038 013 L
A037 001 S
P004 011 L
A038 007 S
P001 008 L
A025 001 S
A037 013 L
A025 002 S
A037 016 L
A038 012 L
A039 008 S
P005 003 L
A040 002 S
P001 014 L
A026 008 S
A039 012 L
A026 007 S
A040 013 L
A039 009 L
A039 001 S
P004 011 L
A040 007 S
P001 014 L
A026 001 S
A039 013 L
A026 002 S
A040 012 L
A039 016 L
A041 008 S
P005 007 L
A042 002 S

P003 008 L
A017 008 S
A041 012 L
A017 007 S
A041 009 L
A042 013 L
A041 001 S
P005 008 L
A042 007 S
P003 008 L
A017 001 S
A041 013 L
A017 002 S
A041 016 L
A042 012 L
A043 008 S
P005 007 L
A044 002 S
P003 014 L
A018 008 S
A043 012 L
A018 007 S
A044 013 L
A043 009 L
A043 001 S
P005 011 L
A044 007 S
P003 014 L
A018 001 S
A043 013 L
A018 002 S
A044 012 L
A043 016 L
A045 008 S
P005 008 L
A046 002 S
P002 002 L
A027 008 S
A045 012 L
A027 007 S
A045 009 L
A046 013 L
A045 001 S
P005 016 L
A046 007 S
P002 002 L
A027 001 S
A045 013 L
A027 002 S
A045 016 L

A046 012 L
A047 008 S
P006 007 L
A048 002 S
P002 004 L
A028 008 S
A047 012 L
A028 007 S
A048 013 L
A047 009 L
A047 001 S
P005 015 L
A048 007 S
P002 004 L
A028 001 S
A047 013 L
A028 002 S
A048 012 L
A047 016 L
A049 008 S
P006 004 L
A050 002 S
P002 006 L
A019 008 S
A049 012 L
A006 009 L
A019 007 S
A049 009 L
A050 013 L
A049 001 S
P006 002 L
A050 007 S
P002 006 L
A019 001 S
A049 013 L
A006 008 L
A019 002 S
A049 016 L
A050 012 L
A051 008 S
P006 006 L
A052 002 S
P002 008 L
A020 008 S
A051 012 L
A020 007 S
A052 013 L
A051 009 L
A006 007 L
A051 001 S

P006 003 L
A052 007 S
P002 008 L
A020 001 S
A051 013 L
A020 002 S
A052 012 L
A051 016 L
A053 008 S
P005 011 L
A054 002 S
P002 012 L
A029 008 S
A053 012 L
A029 007 S
A053 009 L
A054 013 L
A053 001 S
P005 015 L
A054 007 S
P002 012 L
A029 001 S
A053 013 L
A029 002 S
A053 016 L
A054 012 L
A055 008 S
P005 012 L
A056 002 S
P002 014 L
A030 008 S
A055 012 L
A030 007 S
A056 013 L
A055 009 L
A055 001 S
P006 011 L
A056 007 S
P002 014 L
A030 001 S
A055 013 L
A030 002 S
A056 012 L
A055 016 L
A057 008 S
P006 015 L
A058 002 S
P003 002 L
A021 008 S
A057 012 L

A021 007 S
A057 009 L
A058 013 L
A057 001 S
P006 007 L
A058 007 S
P003 002 L
A021 001 S
A057 013 L
A021 002 S
A057 016 L
A058 012 L
A059 008 S
P006 008 L
A060 002 S
P003 004 L
A022 008 S
A059 012 L
A022 007 S
A060 013 L
A059 009 L
A059 001 S
P006 003 L
A060 007 S
P003 004 L
A022 001 S
A059 013 L
A022 002 S
A060 012 L
A059 016 L
P013 015 S
A007 016 L
A009 007 S
A023 012 L
A061 007 S
A023 014 L
A010 007 S
A024 012 L
A062 007 S
A024 014 L
A063 007 S
A015 012 L
A064 007 S
A015 014 L
A065 007 S
A016 012 L
A066 007 S
A016 014 L
A067 007 S
A025 012 L

A068 007 S
A025 014 L
A069 007 S
A026 012 L
A070 007 S
A026 014 L
A071 007 S
A017 012 L
A072 007 S
A017 014 L
A073 007 S
A018 012 L
A074 007 S
A018 014 L
A011 007 S
A027 012 L
A012 007 S
A027 014 L
A013 007 S
A028 012 L
A075 007 S
A028 014 L
A076 007 S
A019 012 L
A077 007 S
A019 014 L
A078 007 S
A020 012 L
A079 007 S
A020 014 L
A080 007 S
A029 012 L
A081 007 S
A029 014 L
A082 007 S
A030 012 L
A083 007 S
A030 014 L
A084 007 S
A021 012 L
A085 007 S
A021 014 L
A086 007 S
A022 012 L
A087 007 S
A022 014 L
P012 007 S
A088 016 L
P010 008 S
A009 016 L

P010 004 S
A061 016 L
P010 007 S
A010 016 L
P015 003 S
A062 016 L
P010 003 S
A063 016 L
P010 015 S
A064 009 L
P006 016 S
A065 016 L
P010 011 S
A066 016 L
P011 007 S
A067 016 L
P011 010 S
A068 016 L
P015 004 S
A069 016 L
P011 011 S
A070 016 L
P011 003 S
A070 009 L
P011 013 S
A072 016 L
P011 015 S
A073 014 L
P011 012 S
A074 014 L
P010 012 S
A011 016 L
P011 006 S
A012 016 L
P013 008 S
A013 016 L
P011 008 S
A075 016 L
P012 008 S
A076 016 L
P013 011 S
A077 016 L
P013 012 S
A078 016 L
P014 016 S
A079 016 L
P013 007 S
A080 016 L
P013 004 S
A081 016 L

P013 015 S
A082 016 L
P013 016 S
A083 016 L
P013 011 S
A084 016 L
P013 004 S
A085 016 L
P012 004 S
A086 016 L
P012 003 S
A087 016 L
A088 001 S
A077 012 L
A008 011 L
A089 007 S
A076 013 L
A014 011 L
A014 013 L
A007 008 L
A089 008 S
A078 013 L
A080 013 L
A081 013 L
A082 013 L
A083 013 L
A089 009 S
A084 013 L
A085 013 L
A086 013 L
A087 013 L
P014 004 S
A089 011 L
P010 014 S
A066 011 L
P011 007 S
A067 011 L
P011 016 S
A068 011 L
P015 003 S
A069 011 L
P011 015 S
A070 012 L
P011 002 S
A071 011 L
A090 008 S
A072 011 L
P007 004 S
A091 010 L
A091 012 S

A090 012 L
A092 002 S
A073 011 L
P007 006 S
A093 016 L
A093 014 S
A092 014 L
A092 008 S
A074 011 L
P007 008 S
A093 010 L
A093 012 S
A092 012 L
A094 002 S
A011 013 L
P007 012 S
A095 016 L
A095 014 S
A094 014 L
A094 008 S
A012 011 L
P007 014 S
A095 010 L
A095 012 S
A094 012 L
A096 002 S
A013 013 L
P007 010 S
A097 016 L
A097 014 S
A096 014 L
A096 008 S
A075 011 L
P007 016 S
A097 010 L
A097 012 S
A096 012 L
A098 002 S
A076 011 L
P008 004 S
A099 016 L
A099 014 S
A098 014 L
A098 008 S
A077 011 L
P009 002 S
A099 010 L
A099 012 S
A098 012 L
A100 002 S

A078 011 L
P008 006 S
A101 016 L
A101 014 S
A100 014 L
A100 008 S
A079 011 L
P008 016 S
A101 010 L
A101 012 S
A100 012 L
P013 008 S
A080 012 L
P012 015 S
A081 012 L
P014 007 S
A082 011 L
P013 016 S
A083 012 L
P013 012 S
A084 012 L
P013 007 S
A085 012 L
P012 011 S
A086 012 L
P012 012 S
A087 012 L
A089 004 S
A008 001 L
A089 001 S
A067 012 L
A068 012 L
A069 012 L
A070 011 L
A071 012 L
A072 012 L
A073 012 L
A074 012 L
A089 002 S
A011 014 L
A012 012 L
A013 014 L
A075 012 L
A076 012 L
A077 012 L
A078 012 L
A079 012 L
P014 003 S
A066 012 L
A089 013 L

A008 004 S
A080 011 L
A081 011 L
A082 012 L
A083 011 L
A084 011 L
A085 011 L
A086 011 L
A087 011 L
P014 012 S
A008 013 L
A039 004 S
P014 015 L
A041 005 S
P011 003 L
A041 004 S
P011 004 L
A043 005 S
P014 011 L
A043 004 S
P014 012 L
A045 005 S
P014 008 L
A045 004 S
P014 007 L
A047 005 S
P015 007 L
A047 004 S
P015 007 L
A049 005 S
P015 012 L
A049 004 S
P015 015 L
A051 005 S
P015 011 L
A051 004 S
P015 008 L
A053 005 S
P014 015 L
A053 004 S
P014 016 L
A055 005 S
P014 011 L
A055 004 S
P015 004 L
A057 005 S
P015 008 L
A057 004 S
P015 011 L
A059 005 S

P015 016 L
A059 004 S
P015 012 L
A088 004 S
A009 012 L
A010 012 L
A011 012 L
A012 014 L
A013 008 L
A014 008 L
A014 014 L
P012 011 S
A088 013 L
P015 016 S
A008 008 L
A008 005 S
A007 001 L
A102 007 S
A009 009 L
A061 009 L
A010 009 L
A062 009 L
A063 009 L
A064 011 L
A065 009 L
A066 009 L
A102 008 S
A067 009 L
A068 009 L
A069 009 L
A070 009 L
A071 001 L
A072 009 L
A073 009 L
A074 009 L
A102 009 S
A011 009 L
A012 009 L
A013 012 L
A075 009 L
A076 009 L
A077 009 L
A078 009 L
A079 009 L
A102 001 S
A080 009 L
A081 009 L
A082 009 L
A083 009 L
A084 009 L

A085 009 L
A086 009 L
A087 009 L
P012 015 S
A088 009 L
A088 008 S
A102 011 L
A102 013 L
A103 002 S
A009 008 L
P007 006 S
A104 016 L
A104 014 S
A103 014 L
A103 008 S
A061 008 L
P007 008 S
A104 010 L
A104 012 S
A103 012 L
A105 002 S
A010 008 L
P007 004 S
A106 016 L
A106 014 S
A105 014 L
A105 008 S
A062 008 L
P007 010 S
A106 010 L
A106 012 S
A105 012 L
A107 002 S
A063 008 L
P007 014 S
A108 016 L
A108 014 S
A107 014 L
A107 008 S
A064 012 L
P007 016 S
A108 010 L
A108 012 S
A107 012 L
A109 002 S
A065 008 L
P007 012 S
A110 016 L
A110 014 S
A109 014 L

A109 008 S
A066 008 L
P008 002 S
A110 010 L
A110 012 S
A109 012 L
A111 002 S
A067 008 L
P008 006 S
A112 016 L
A112 014 S
A111 014 L
A111 008 S
A068 008 L
P008 008 S
A112 010 L
A112 012 S
A111 012 L
A113 002 S
A069 008 L
P008 004 S
A114 016 L
A114 014 S
A113 014 L
A113 008 S
A070 008 L
P008 012 S
A114 010 L
A114 012 S
A113 012 L
A115 002 S
A071 016 L
P008 002 S
A116 016 L
A116 014 S
A115 014 L
A115 008 S
A072 008 L
P009 004 S
A116 010 L
A116 012 S
A115 012 L
A117 002 S
A073 008 L
P008 012 S
A118 016 L
A118 014 S
A117 014 L
A117 008 S
A074 008 L

P008 014 S
A118 010 L
A118 012 S
A117 012 L
A119 002 S
A011 008 L
P009 006 S
A120 016 L
A120 014 S
A119 014 L
A119 008 S
A012 008 L
P009 012 S
A120 010 L
A120 012 S
A119 012 L
A121 002 S
A013 011 L
P009 008 S
A122 016 L
A122 014 S
A121 014 L
A121 008 S
A075 008 L
P009 004 S
A122 010 L
A122 012 S
A121 012 L
A123 002 S
A076 008 L
P009 014 S
A124 016 L
A124 014 S
A123 014 L
A123 008 S
A077 008 L
P009 016 S
A124 010 L
A124 012 S
A123 012 L
A125 002 S
A078 008 L
P009 010 S
A126 016 L
A126 014 S
A125 014 L
A125 008 S
A079 008 L
P008 016 S
A126 010 L

A126 012 S
A125 012 L
A127 002 S
A080 008 L
P008 014 S
A128 016 L
A128 014 S
A127 014 L
A127 008 S
A081 008 L
P009 002 S
A128 010 L
A128 012 S
A127 012 L
A129 002 S
A082 008 L
P009 006 S
A130 016 L
A130 014 S
A129 014 L
A129 008 S
A083 008 L
P009 008 S
A130 010 L
A130 012 S
A129 012 L
A131 002 S
A084 008 L
P009 012 S
A132 016 L
A132 014 S
A131 014 L
A131 008 S
A085 008 L
P009 014 S
A132 010 L
A132 012 S
A131 012 L
A133 002 S
A086 008 L
P009 010 S
A134 016 L
A134 014 S
A133 014 L
A133 008 S
A087 008 L
P009 016 S
A134 010 L
A134 012 S
A133 012 L

A014 005 S
A077 013 L
A014 004 S
A079 013 L
P012 016 S
A088 012 L
A088 005 S
A135 011 L
A135 013 L
A135 007 S
A009 001 L
A061 001 L
A010 001 L
A062 001 L
A063 001 L
A064 008 L
A065 001 L
A066 001 L
A135 008 S
A067 001 L
A068 001 L
A069 001 L
A070 001 L
A071 008 L
A072 001 L
A073 013 L
A074 013 L
A135 009 S
A011 001 L
A012 001 L
A013 001 L
A075 001 L
A076 001 L
A077 001 L
A078 001 L
A079 001 L
A135 001 S
A080 001 L
A081 001 L
A082 001 L
A083 001 L
A084 001 L
A085 001 L
A086 001 L
A087 001 L

APPENDIX B

```
BEGIN
%
%       THE CHANNEL ROUTING PROGRAM WHICH FOLLOWS WAS DEVELOPED AT THE
%   UNIVERSITY OF ILLINOIS BY JIM STEVENS.  THE DESIGN OF THIS PRO-
%   GRAM IS THE SUBJECT OF CHAPTER FIVE OF THE PHD THESIS ENTITLED
%   HEURISTIC TECHNIQUES FOR THE AUTOMATED DESIGN OF PRINTED CIRCUIT
%   BOARDS.  BRIEFLY THE ALGORITHM PROCEEDS BY DIVIDING EACH CONNECTION
%   INTO THREE WIRE SEGMENTS, TWO HORIZONTAL AND ONE VERTICAL.  THESE
%   WIRE SEGMENTS ARE THEN ASSIGNED TO SPACES ON THE BOARD WHICH ARE
%   IMPLEMENTED AS TWO ARRAYS, VERTS FOR VERTICAL WIRES AND HORZ FOR
%   HORIZONTAL WIRES.  THE PROGRAM ASSUMES THAT THE BOARD IS COMPOSED
%   OF ELEVEN ROWS OF FIFTEEN 16 PIN DIL IC PACKAGES.  THE SPACING
%   BETWEEN ROWS AND COLUMNS OF PACKAGES IS DEPENDANT ON THE VARIABLES
%   VS AND HS.  FINAL POSITIONS OF THE WIRE SEGMENTS IS DETERMINED BY
%   THE CHANNEL ASSIGNMENT ALGORITHM.
%
%       EACH HORIZONTAL WIRE SEGMENT IS REPRESENTED AS ONE 47 BIT
%   WORD WITH THE FOLLOWING FIELDS.  VERTICAL WIRE SEGMENTS ARE
%   REPRESENTED WITH THE SAME TYPE OF WORD ONLY THE END POINTS ARE
%   CALLED TOP AND BOTTOM.
%
%   BITS          USAGE
%
%   0 TO 5        OFFSET OF RIGHT END POINT WITHIN SPACE
%   6 TO 10       VERTICAL SPACE OF RIGHT END POINT
%   11 TO 16      OFFSET OF LEFT END POINT WITHEN SPACE
%   17 TO 21      VERTICAL SPASE OF LEFT END PBINT
%   22 TO 30      POINTER TO VERTICAL WIRE SEGMENT CONNECTED TO THE
%                 RIGHT END OF THIS WIRE SEGMENT
%   31 TO 39      POINTER TO VERTICAL WIRE SEGMENT CONNECTED TO THE
%                 LEFT END OF THIS WIRE SEGMENT
```

```
%    40            TAG TO INDICATE FINAL POSITIONING OF THIS WIRE SEG
%    41  TO 46     NUMBER OF THE CHANNEL THIS WIRE SEGMENT IS ASSIGNED
%
    FILE PRINT 15(2,17);
  FILE CARDS (2,10);
  FILE WIRLIST DISK SERIAL "RRRRRR"/"WIRLIST" (2,10,30);
    INTEGER RIT,LFT,TOP,BOT,RITOFF,TB1,TB2,IP,BIG,IND,BIGP,
            LFTOFF,TOPOFF,BOTOFF,I,J,RPTR,
            J1,J2,HS,HSH,VS,VSH,PTR,SPAC ,
            T1,I2,VPH,HPH,VP,HP,FIRST,LAST,PREUSE,DUMB,
            CHANNEL,BEST,RLIM,BJ,K,RLIM,IT,JJ,SPEC,
            LPTR,TPTR,BPTR,TEMP;
    INTEGER FIRSTP,LASTP,RT,LT,RTOFF,LTOFF,RTPTR,LTPTR;
     ALPHA  ARRAY HORS[0:11,0:255],HSPTR[0:11],
            VERTS[0:14,0:255],VSPTR[0:15];
     ALPHA  ARRAY HORP[0:23,0:255],HPPTR[0:23],
            VERTP[0:14,0:255],VPPTR[0:14];
    FORMAT  FMT(4I3),CHECK((X2,2I5/)),NFM(X2,15I5),
           NFMT(2I8),ERFMT("POINTER ERROR",O  ,"DOES NOT POINT TO",I6),
           CHFMT(I6,"   CHANNELS USED IN SPACE  ",I5/);
    LABEL NEXTV,NEXTH,ERROR,EDIT;
    LABEL  EOF,MORE;
    LABEL  RITE,RITO;
    LABEL  ERROR1;
  ALPHA FILNAME;
  FORMAT CARDFMT(A6);
    READ(CARDS,CARDFMT,FILNAME);
    FILL WIRLIST WITH   FILNAME&"R"[01:5:6];
    WRITE(PRINT,CARDFMT,FILNAME);
         VSH:=(VS:=9) DIV 2;
       VPH:=(VP:=7) DIV 2 ;
         HSH:=(HS:=9 ) DIV 2;
%
```

```
*         READ FROM WIRELIST. FOUR ENTRIES FOR EACH CONNECTION.  FIRST
*    GIVES THE COORDINATES OF THE ROW AND COLUMN OF ONE PACKAGE TO BE
*    CONNECTED AT  THE PIN SPECIFIED BY FIRSTP. LAST GIVES THE POSITION
*    OF THE OTHER PACKAGE TO BE CONNECTED AT PIN LASTP.  EDGE CONNECTOR
*    PINS OF THE BOARD APPEAR AS PACKAGES IN ROW ZERO.
*
MORE:  READ(WIRLIST,FMT,FIRST,FIRSTP,LAST,LASTP)[EOF];
         VSPTR[15]:=255;
         TB1:=TB2:=0;
         I1:=FIRST.[7:4];
         J1:=FIRST.[3:4];
         I2:=LAST.[7:4];
         J2:=LAST.[3:4];
     IF J1=J2 THEN
*
*    BOTH PACKAGES IN ONE COLUMN.
*
          IF VSPTR[J1+1] LSS VSPTR[J1] THEN
           BEGIN
            J:=RT:=RIT:=J1+1;
            LT:=LFT:=J1;
            RTPTR:=RPTR:=VSPTR[J];
            LTPTR:=LPTR:=511;
            RITOFF:=RTOFF:=VSH;
            IF LASTP GTR 8 THEN BEGIN
             LFTOFF:=VS+17-LASTP;
            TB2:=1;
            END ELSE
             LFTOFF:=VS+LASTP;
            IF FIRSTP GTR  8 THEN BEGIN
             LTOFF:=VS+ 17-FIRSTP;
             TB1:=1;
```

```
            END ELSE
             LTOFF:=VS+FIRSTP;
          END ELSE
        BEGIN
         J:=RT:=RIT:=J1;
              LT:=LFT:=J1;
         RTPTR:=RPTR:=511;
         LTPTR:=LPTR:=VSPTR[J];
         LTOFF:=LFTOFF:=VSH;
         IF LASTP GTR 8 THEN BEGIN
          RTTOFF:=VS+16-LASTP;
          TR2:=1;
         END ELSE
          RTTOFF:=VS+LASTP-1;
         IF FIRSTP GTR 8 THEN BEGIN
          RTOFF:=VS+16-FIRSTP;
          TR1:=1;
         END ELSE
          RTOFF:=VS+FIRSTP-1;
       END;
            IF J1 GTR J2 THEN
%
%    GENERAL CONNECTION
%
         BEGIN
%
%        THE VERTICAL WIRE SEGMENT FOR EACH CONNECTION IS ASSIGNED
%    TO THE VERTICAL SPACE WHICH HAS THE FEWEST WIRE SEGMENTS ALREADY
%    ASSIGNED TO IT AND LIES BETWEEN THE COLUMNS OF THE TWO PACKAGES
%    BEING CONNECTED.
%
             BIG:=255;
```

```
            FOR IND:=J2+1 STEP 1 UNTIL J1 DO
             IF VSPTR[IND] LSS BIG THEN BEGIN
              BIG:=VSPTR[IND];
              BIGP:=IND;
             END;
           J:=RIT:=BIGP;
           LFT:=J2;
           RTOFF:=VSH;
         IF LASTP>8 THEN BEGIN
                        LFTOFF:=VS+17-LASTP;
             TR2:=1;
           END ELSE
             LFTOFF:=VS+LASTP;
           RPTR:=VSPTR[J];
           LPTR:=511;
         RT:=J1;
         LT:=J;
          IF FIRSTP>8 THEN BEGIN
             RTOFF:=VS+16-FIRSTP;
             TR1:=1;
           END ELSE
             RTOFF:=VS+FIRSTP-1;
         LTOFF:=VSH;
         LTPTR:=VSPTR[J];
         RTPTR:=511;
           END;
          IF J1 LSS J2 THEN
          BEGIN
%
%       THE VERTICAL WIRE SEGMENT FOR EACH CONNECTION IS ASSIGNED
%   TO THE VERTICAL SPACE WHICH HAS THE FEWEST WIRE SEGMENTS ALREADY
%   ASSIGNED TO IT AND LIES BETWEEN THE COLUMNS OF THE TWO PACKAGES
```

```
%    BEING CONNECTED.
%
          BIG:=255;
          FOR IND :=J1+1 STEP 1 UNTIL J2 DO
           IF VSPTR[IND] LSS BIG THEN BEGIN
            BIG:=VSPTR[IND];
            BIGP:=IND;
           END;
        J:=LFT:=BIGP;
        RIT:=J2;
      IF LASTP>8 THEN BEGIN
            RITOFF:=VS+16-LASTP;
            TB2:=1;
          END ELSE
            RITOFF:=VS+LASTP-1;
        LFTOFF:=VSH;
        RPTR:=511;
        LPTR:=VSPTR[J];
      RT:=J;
      LT:=J1;
      IF FIRSTP>8 THEN BEGIN
            LTOFF:=VS+17-FIRSTP;
            TB1:=1;
          END ELSE
            LTOFF:=VS+FIRSTP;
      RTOFF:=VSH;
      RTPTR:=VSPTR[J];
      LTPTR:=511;
       END;
      IF I1≥I2 THEN
       BEGIN
        I:=BOT:=2×I2+TB2;
        IP:=TOP:=2×I1+TB1;
```

```
            TOPOFF:=   HSH;
          BOTOFF:=   HSH;
          TPTR:=HPPTR[TOP];
            BPTR:=HPPTR[I];
           END ELSE
           BEGIN
            I:=TOP:=2*I2+TB2;
            IP:=BOT:=2*I1+TB1;
          TOPOFF:=   HSH;
            BOTOFF:=   HSH;
            TPTR:=HPPTR[I];
          BPTR:=HPPTR[BOT];
           END;
           IF I1=I2 THEN
%
%    BOTH PINS IN SAME PIN ROW. NO VERTICAL WIRE IS USED.
%
            BEGIN
            IF J1≥J2 THEN
            BEGIN
             RTOFF:=RTOFF;
             RPTR:=RTPTR;
             END ELSE
            BEGIN
             LFT:=LT;
             LFTOFF:=LTOFF;
             LPTR:=LTPTR;
            END;
%
%    STORE WORD FOR HORIZONTAL ONLY, BOTH PINS IN SAME PIN ROW.
%
          HORP[TOP,HPPTR[TOP]]:=
               RTOFF & LPTR[39:8:9] &
```

```
                RPTR[30:8:9] & LFT[21:4:5] &
                LFTOFF[16:5:6] & RIT[10:4:5];
            HPPTR[TOP]:=HPPTR[TOP]+1;
              GO TO MORE;
             END;
%
%    STORE WORDS FOR GENERAL CONNECTION.
%
              HORP[I,HPPTR[I]]:=RTOFF        &LPTR[39:8:9]
                   &RPTR[30:8:9]&LFT[21:4:5]&LFTOFF[16:5:6]&RIT[10:4:5];
              HPPTR[I]:=HPPTR[I]+1;
            HORP[IP, HPPTR[IP]]:=RTOFF & LTPTR[39:8:9] & RTPTR[30:8:9] &
              LT[21:4:5] & LTOFF[16:5:6] & RT[10:4:5];
            HPPTR[IP]:=HPPTR[IP]+1;
              VERTS[J,VSPTR[J]]:=TOPOFF        &BPTR[39:8:9]&TPTR[30:8:9]
                    &BOT[21:4:5]&BOTOFF[16:5:6]&TOP[10:4:5];
              VSPTR[J]:=VSPTR[J]+1;
             GO TO MORE;
          EOF;
%
%    END SPACE ASSIGNMENT
%
          WRITE(PRINT,NEW,FOR I:=0 STEP 1 UNTIL 14 DO VSPTR[I],
                FOR I:=0 STEP 1 UNTIL 23 DO HPPTR[I]);
%
%    BEGIN CHANNEL ASSIGNMENT
%
            CHANNEL:=5;
            FOR K:=0 STEP 1 UNTIL 14 DO
             BEGIN
            IF VSPTR[K]=0 THEN
             BEGIN
              CHANNEL:=0;
```

```
            GO TO RITE;
            END;
          IF CHANNEL GTR (2*VS+7) THEN        %FAILURE HAS OCCURED.
           CHANNEL:=VS+1
          ELSE IF CHANNEL GTR (VS+7) THEN
           CHANNEL:=CHANNEL-(VS+6)
          ELSE CHANNEL:=1;
            BEST:=0;
            BLIM:=2047;
            FOR I:=0 STEP 1 UNTIL VSPTR[K]-1 DO
             BEGIN
%
%          FIND THE WIRE SEGMENT WITH THE LARGEST UPPER BOUND SMALLER
%    THAN THE LOWER BOUND OF THE PREVIOUS WIRE SEGMENT.
%
  NEXTV:      FOR J:=0 STEP 1 UNTIL VSPTR[K]-1 DO
               IF VERTS[<,J].[40:1]=0 THEN
                IF VERTS[<,J].[10:11]<BLIM THEN
                 IF VERTS[<,J].[10:11]>BEST THEN
                  BEGIN
                   BEST:=VERTS[<,J].[10:11];
                   BJ:=J;
                  END;
              IF BEST=0 THEN
               BEGIN
                BLIM:=2047;
                CHANNEL:=CHANNEL+1;
                GO TO NEXTV;
               END;
             IF PTR NEQ 511 THEN
               IF HORP[SPAC,PTR].[10:5]=K AND
                  HORP[SPAC,PTR].[30:9]=BJ THEN
```

```
            IF HORP[SPAC,PTR],[16:6] GEQ CHANNEL THEN
              HORP[SPAC,PTR]:=HORP[SPAC,PTR]&CHANNEL[5:5:6]
            ELSE                    %REVERSAL HAS OCCURED.
              HORP[SPAC,PTR]:=HORP[SPAC,PTR]&TEMP[39:30:9]
                  &TEMP[30:39:9]&TEMP[21:10:5]&TEMP[10:21:5]
                  &(DUMR:=TEMP,[16:6]+1)[5:5:6]^CHANNEL[16:5:6] ELSE
            IF HORP[SPAC,PTR],[21:5]=K AND
               HORP[SPAC,PTR],[39:9]=BJ THEN
            IF HORP[SPAC,PTR],[ 5:6] GEQ CHANNEL THEN
              HORP[SPAC,PTR]:=HORP[SPAC,PTR]&CHANNEL[16:5:6]
            ELSE                    %REVERSAL HAS OCCURED.
              HORP[SPAC,PTR]:=HORP[SPAC,PTR]&TEMP[39:30:9]
                  &TEMP[30:39:9]&TEMP[21:10:5]&TEMP[10:21:5]
                  &(DUMR:=TEMP,[5:6]+1)[16:5:6]&CHANNEL[ 5:5:6] ELSE
            BEGIN
ERROR:          WRITE(PRINT,ERFMT,HORP[SPAC ,PTR],K);
                GO TO EOUT;
              END;
          END;
RITE:     WRITE(PRINT,CHFMT,CHANNEL,K);
        END;
       CHANNEL:=1;
       FOR K:=0 STEP 1 UNTIL 23 DO
        BEGIN
         IF HPPTR[K]=0 THEN
          BEGIN
           CHANNEL:=0;
           GO TO RITE;
          END;
        VERTS[K,BJ1:=VERTS[K,BJ1&1[40:0:1]&CHANNEL[45:5:6]];
        REST:=0;
        BLIM:=VERTS[K,BJ],[21:11];
```

```
*
*       REPOSITION THE END POINT OF THE HORIZONTAL WIRE SEGMENT WHICH
*   CONNECTS TO THIS WIRE SEGMENT TO CORESPOND TO THIS CHANNEL
*   ASSIGNMENT. REVERSAL OF THE HORIZONTAL WIRE SEGMENT IS POSSIBLE
*
      PTR:=VERTS[K,BJ].[30:9];
      SPAC:=VERTS[K,BJ].[10:5];
      TEMP:=HORP[SPAC,PTR];
      IF PTR NEQ 511 THEN
        IF HORP[SPAC,PTR].[10:5]=K AND
           HORP[SPAC,PTR].[30:9]=BJ THEN
        IF HORP[SPAC,PTR].[16:6] GEQ CHANNEL THEN
          HORP[SPAC,PTR]:=HORP[SPAC,PTR]&CHANNEL[5:5:6]
        ELSE                    %REVERSAL HAS OCCURED.
          HORP[SPAC,PTR]:=HORP[SPAC,PTR]&TEMP[39:30:9]
              &TEMP[30:39:9]&TEMP[21:10:5]&TEMP[10:21:5]
              &(OUMR:=TEMP.[16:6]+1)[5:5:6]&CHANNEL[16:5:6] ELSE
        IF HORP[SPAC,PTR].[21:5]=K AND
           HORP[SPAC,PTR].[39:9]=BJ THEN
        IF HORP[SPAC,PTR].[ 5:6] GEQ CHANNEL THEN
          HORP[SPAC,PTR]:=HORP[SPAC,PTR]&CHANNEL[16:5:6]
        ELSE                    %REVERSAL HAS OCCURED.
          HORP[SPAC,PTR]:=HORP[SPAC,PTR]&TEMP[39:30:9]
              &TEMP[30:39:9]&TEMP[21:10:5]&TEMP[10:21:5]
              &(OUMR:=TEMP.[5:6]+1)[16:5:6]&CHANNEL[ 5:5:6] ELSE
            GO TO ERROR;
      PTR:=VERTS[K,BJ].[39:9];
      SPAC:=VERTS[K,BJ].[21:5];
      TEMP:=HORP[SPAC,PTR];
        IF (K MOD 2)=0 THEN
         IF CHANNEL GTR 19 THEN     %FAILURE HAS OCCURED.
          CHANNEL:=15
         ELSE IF CHANNEL GTR 5 THEN
```

```
          CHANNEL:=CHANNEL-4
         ELSE CHANNEL:=1
        ELSE
         IF CHANNEL GTR 19 THEN     %FAILURE HAS OCCURED.
          CHANNEL:=6
         ELSE IF CHANNEL GTR 14 THEN
          CHANNEL:=CHANNEL-13
         ELSE CHANNEL:=1;
          BEST:=0;
          RLIM:=2047;
          FOR I:=0 STEP 1 UNTIL HPPTR[K]-1 DO
           BEGIN
%
%          FIND THE WIRE SEGMENT WITH THE  RIGHTMOST RIGHTHAND BOUND
%    TO THE LEFT OF THE LEFTHAND BOUND OF THE PREVIOUS WIRE SEGMENT.
%
  NEXTH;     FOR J:=0 STEP 1 UNTIL HPPTR[K]-1 DO
              IF HORP[K,J].[40:1]=0 THEN
               IF HORP[K,J].[10:11]<RLIM THEN
                IF HORP[K,J].[10:11]>BEST THEN
                 BEGIN
                  BEST:=HORP[K,J].[10:11];
                  BJ:=J;
                 END;
             IF BEST=0 THEN
              IF HPPTR[K]≠0 THEN
              BEGIN
               RLIM:=2047;
               CHANNEL:=CHANNEL+1;
               GO TO NEXTH;
              END;
             HORP[K,BJ]:=HORP[K,BJ]&1[40:0:1]&CHANNEL[46:5:6];
```

```
            RLIM:=HORP[K,BJ].[21:11];
            REST:=0;
%
%       REPOSITION THE END POINT OF THE VERTICAL WIRE SEGMENT WHICH
%   CONNECTS TO THIS WIRE SEGMENT TO CORESPOND TO THIS CHANNEL ASSIGN-
%   MENT. NO REVERSALS OF VETICAL WIRE SEGMENTS ARE POSSIBLE
%
          PTR:=HORP[K,BJ].[30:9];
          SPAC:=HORP[K,BJ].[10:5];
          IF PTR NEQ 511 THEN
            IF VERTS[SPAC,PTR].[10:5]=K THEN
              VERTS[SPAC,PTR]:=VERTS[SPAC,PTR]&CHANNEL[5:5:6] ELSE
            IF VERTS[SPAC,PTR].[21:5]=K THEN
              VERTS[SPAC,PTR]:=VERTS[SPAC,PTR]&CHANNEL[16:5:6]
            ELSE GO TO ERROR1;
          PTR:=HORP[K,BJ].[39:9];
          SPAC:=HORP[K,BJ].[21:5];
          IF PTR NEQ 511 THEN
            IF VERTS[SPAC,PTR].[10:5]=K THEN
              VERTS[SPAC,PTR]:=VERTS[SPAC,PTR]&CHANNEL[5:5:6] ELSE
            IF VERTS[SPAC,PTR].[21:5]=K THEN
              VERTS[SPAC,PTR]:=VERTS[SPAC,PTR]&CHANNEL[16:5:6]
            ELSE BEGIN
ERROR1:       WRITE(PRINT,ERFMT,VERTS[SPAC,PTR],K);
              GO TO EOUT;
            END;
          END;
RITO:     WRITE(PRINT,CHFMT,CHANNEL,K);
          END;
  EOUT:
    END.
```

LIST OF REFERENCES

1. G. H. Barnes, R. M. Brown, M. Kato, D. J. Kuck, D. L. Slotnick and R. A. Stokes, "The ILLIAC IV Computer," IEEE Transactions on Computers, C-17, No. 8, (1968), 746.

2. D. L. Slotnick, "The Fastest Computer," Scientific American, Vol. 224, No. 2, (February 1971), 76-87.

3. M. A. Breuer, "General survey of design automation of digital computers," Proc. IEEE, 54, No. 12, (December 1966), 1708-1721.

4. J. Munkres, "Algorithms for the assignment and transportation problems," J. SIAM, 5, (1957), 32-38.

5. E. L. Lawler, "The quadratic assignment problem," Management Science 9, (1963), 586-599.

6. P. C. Gilmore, "Optimal and suboptimal algorithms for the quadratic assignment problem," J. SIAM, 10, No. 2, (June 1962), 305-313.

7. R. C. Prim, "Shortest connection networks and some generalizations," Bell Sys. Tech. J., 36, (November 1957), 1389-1401.

8. H. Loberman and A. Weinberger, "Formal procedures for connecting terminals with a minimum total wire length," J. ACM 4, (October 1957), 428-437.

9. R. Courant and H. Robbins, What is Mathematics?, Oxford University Press, New York, 1941.

10. M. Hanan, "On Steiner's problem with rectilinear distance," J. SIAM, 14, (1966), 255-265.

11. M. Hanan and J. M. Kurtzberg, Placement techniques, Design Automation of Digital Systems: Theory and Techniques, M. Breuer, ed., Prentice-Hall, New York, 1972.

12. J. M. Kurtzberg, "Backboard wiring algorithms for the placement and connection order problems," Burroughs Report TR 60-40, June 28, 1960.

13. J. M. Kurtzberg, "On approximation methods for the assignment problem," J. ACM, 9, No. 4, (October 1962), 419-439.

14. J. M. Kurtzberg, "Algorithms for backplane formation," in Micro-electronics in Large Systems, Spartan Books, 1965, 51-76.

15. R. L. Gamblin, M. Q. Jacobs and C. J. Tunis, "Automatic packaging of miniaturized circuits," in Advances in Electronic Circuit Packaging, Vol. 2, G. A. Walker, ed., New York: Plenum, 1962, 219-232.

16. L. Steinberg, "The backboard wiring problem: A placement algorithm," SIAM Review, 3 No. 1, (January 1961), 37-50.

17. R. A. Rutman, "An algorithm for placement of interconnected elements based on minimum wire length," Proc. 1964 SJCC, 477-491.

18. F. S. Hillier and M. M. Conners, "Quadratic assignment problem algorithms and the location of indivisible location," Management Science, 13, No. 1, (September 1966), 42-57.

19. G. W. Graves and A. B. Winston, "An algorithm for the quadratic assignment problem," Management Science, 17, No. 7, (March 1970), 453-471.

20. C. Y. Lee, "An algorithm for path connections and its applications," IRE Transactions on Electronic Computers, (September 1961), 346-365.

21. D. W. Hightower, "A solution to line-routing problems on the continuous plane," Proc. 1969 Design Automation Workshop, 1-24.

22. K. Mikami and K. Tabuchi, "A computer program for optimal routing of printed circuit conductors," Proceedings of the IFIP Congress, 2, 1968, 1475.

23. R. B. Hitchcock, "Cellular wiring and the cellular modeling technique, technique," Proc. 1969 Design Automation Workshop, 25-41.

24. S. Heiss, "A path connection algorithm for multilayer boards," Proc. 1968 Design Automation Workshop.

25. S. Ackers, Routing techniques, Design Automation of Digital Systems: Theory and Techniques, M. Breuer, ed., Prentice-Hall, New York, 1972.

26. S. E. Lass, "Automated printed circuit routing with a stepping aperture," Comm. of ACM, 12, No. 5, (May 1969), 262-265.

27. A. Hashimoto and J. E. Stevens, Jr., "Wire routing by optimizing channel assignment within large apertures," Proc. 1971 Design Automation Workshop.

28. A. Hashimoto and J. E. Stevens, Jr., "Path cover of acyclic graphs," ILLIAC IV Document No. 239, December 24, 1970.

29. R. G. Garside and T. A. Nicholson, "Permutation procedure for the backboard wiring problem," Proc. IEE London, (January 1968), 27-30.

30. L. C. Abel, "On the automated layout of multi-layer planar wiring and a related graph coloring problem," Coordinated Science Laboratory Report R-546, University of Illinois at Urbana-Champaign, January 1972.

UNCLASSIFIED

Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author)

Center for Advanced Computation
Univeristy of Illinois at Urbana-Champaign
Urbana, Illinois 61801

2a. REPORT SECURITY CLASSIFICATION

UNCLASSIFIED

2b. GROUP

3. REPORT TITLE

Fast Heuristic Techniques for Placing and Wiring Printed Circuit Boards

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

Doctoral Dissertation

5. AUTHOR(S) (First name, middle initial, last name)

James E. Stevens, Jr.

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
October 1972	128	30

8a. CONTRACT OR GRANT NO.	9a. ORIGINATOR'S REPORT NUMBER(S)
DAHC04 72-C-0001	CAC Document No. 58
b. PROJECT NO.	
c. ARPA Order No. 1899	9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)
d.	

10. DISTRIBUTION STATEMENT

Copies may be requested from the address given in (1) above.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
	U.S. Army Research Office-Durham Duke Station, Durham, North Carolina

13. ABSTRACT

A comparison of existing placement techiques is presented. Large realistic printed circuit board problems are introduced and used to make comparisons. The value of different placements is measured by the total wire length generated as well as by the results of actually routing the connections on the board.

A new wire routing algorithm called Channel Routing, is proposed. The Channel Routing algorithm is faster than existing routing techniques and can handle very large circuit board problems. The results of implementing the basic Channel Routing algorithms are presented. These results are also used to compare the effectiveness of the different placement algorithms.

DD FORM 1 NOV 68 1473

UNCLASSIFIED

Security Classification

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Fast Heuristic Techniques						
Printed Circuit Boards						
ILLIAC IV Design Problem						
Design Automation						
Placement						
Routing						

BIBLIOGRAPHIC DATA SHEET	1. Report No. UIUCDCS-R-72-558	2.	3. Recipient's Accession No.

4. Title and Subtitle

Fast Heuristic Techniques for Placing and Wiring Printed Circuit Boards

5. Report Date: October 1972

6.

7. Author(s): James Edward Stevens, Jr.

8. Performing Organization Rept. No.

9. Performing Organization Name and Address

Center for Advanced Computation
University of Illinois at Urbana-Champaign
Urbana, Illinois 61801

10. Project/Task/Work Unit No.

11. Contract/Grant No.: DAHC04 72-C-0001

12. Sponsoring Organization Name and Address

U. S. Army Research Office - Durham
Duke Station
Durham, North Carolina

13. Type of Report & Period Covered: Doctoral Dissertation

14.

15. Supplementary Notes

None

16. Abstracts

A comparison of existing placement techniques is presented. Large realistic printed circuit board problems are introduced and used to make comparisons. The value of different placements is measured by the total wire length generated as well as by the results of actually routing the connections on the board.

A new wire routing algorithm called Channel Routing, is proposed. The Channel Routing algorithm is faster than existing routing techniques and can handle very large circuit board problems. The results of implementing the basic Channel Routing algorithms are presented. These results are also used to compare the effectiveness of the different placement algorithms.

17. Key Words and Document Analysis. 17a. Descriptors

Fast Heuristic Techniques
Printed Circuit Boards
ILLIAC IV Design Problem
Design Automation
Placement
Routing

17b. Identifiers/Open-Ended Terms

17c. COSATI Field/Group

18. Availability Statement	19. Security Class (This Report)	21. No. of Pages
Copies may be obtained from the address in (9) above. Distribution unlimited.	UNCLASSIFIED	128
	20. Security Class (This Page) UNCLASSIFIED	22. Price NC

BOUND BY
HNM

UNIVERSITY OF ILLINOIS-URBANA
510.84IL63C C001
CAC DOCUMENT$URBANA
51-60 1972-73
3 0112 007264143